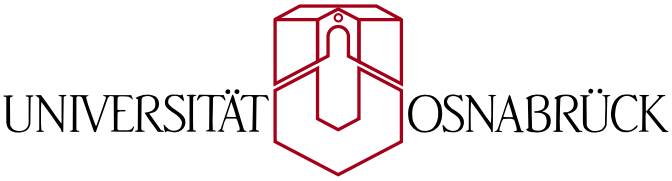


Institut für Informatik

Registration of Historic and Modern Images in Urban Rephotography

Dissertation

zur Erlangung des Doktorgrades (Dr. rer. nat.)
des Fachbereichs Mathematik/Informatik
der Universität Osnabrück

vorgelegt von

Ann-Katrin Becker

Januar 2020

Betreut durch: Prof. Dr. Oliver Vornberger

Abstract

This thesis tackles the challenge of registering modern to historic images in the context of urban rephotography. It aims at automatically identifying stable image features in scenes, which have been exposed to medium to tremendous changes across the years. Instead, the related field of location recognition mainly focuses on illumination and seasonal changes. This work illustrates that common feature descriptors are applicable in the context of historic and modern image matching, while local detectors are not, but most important is the choice of appropriate correspondence filters. It is verified that major structural changes are most challenging for traditional image matching approaches and the methods developed in this work are applicable to challenging image pairs beyond rephotography. Besides, features extracted from Convolutional Neural Networks (CNNs), originally trained for the task of location recognition, show high performance and should be further developed for the specific task of historic to modern image matching. At last, practical developments are presented, including an online portal for presenting and organizing rephotographs as well as an initial version of a mobile application, which supports recovering the original viewpoint of an image.

Contents

Acronyms

BOW Bag-of-Words
BRIEF Binary Robust Independent Elementary Features
BRISK Binary Robust Invariant Scalable Keypoints
CNN Convolutional Neural Network
DGF Disparity Gradient Filter
DGS Disparity Gradient Sum
DoG Difference of Gaussians
FAST Features from Accelerated Segment Test
FLANN Fast Approximate Nearest Neighbour Search
FREAK Fast Retina Keypoint
FV Fisher Vectors
HOG Histogram of Oriented Gradients
K-VLD K-connected VLD-based matching
LIOP Local Intensity Order Pattern
LATCH Learned Arrangements of Three Patch Codes
LMedS Least Median of Squares
MSER Maximally Stable Extremal Regions
ORB Oriented FAST and Rotated BRIEF
SfM Structure from Motion
SIFT Scale-Invariant Feature Transform
SURF Speeded Up Robust Features
SVM Support Vector Machine

Chapter 1

Introduction

1.1 Motivation

Rephotography, also known as repeat photography, is the process of retaking a photograph that is usually much older and considered to be historic. In doing so, the original scene is recaptured from the exact same viewpoint. The resulting picture pair provides a compelling visualization of the respective locations change throughout the years, as shown in Figure 1.1.

The purpose of creating rephotographs is manifold. In ecology they are used to document vegetation and climate change [Clements, 1905; Meyer and Youngs, 2018; Rhemtulla et al., 2002], glacier melting [Gore, 2006] and geological erosion [Hall, 2002]. Sociologists use them for the analysis of urban development and social change [Walker and Leib, 2002]. Other works are simply addressed at interested citizens [Klett et al., 1984; Levere et al., 2004].

However, even for experienced rephotographers recapturing a particular scene is often tedious and time consuming. Common practice still is to estimate the desired viewpoint by eye via manually comparing the scene in front of the camera lens with a printout of the original [Gehrt, 2014; Harrison, 1974; Klett, 2004; Klett et al., 1984; Malde, 1973]. Thereby the photographer needs to distinguish the six degrees of freedom of translation and rotation and additionally keep the effects of camera zoom in mind. Assisted methods utilizing modern computer vision techniques were presented [Bae et al., 2010; Lee et al., 2011; Shrivastava et al., 2011], but they are not publicly accessible. Furthermore, the presented approaches still require a great amount

Figure 1.1: Rephotograph of the main station in Osnabrueck. Left image by Rudolf Lichtenberg (1914), right image by Oliver Vornberger (2013).

of user interaction, as similar points in the historic and modern scene need to be identified manually.

In practice, there are two different approaches which may be pursued to simplify the creation of rephotographs. The first is to support the photographer in the process of recapturing the scene, as Bae et al. [2010] proposed. In this case, the ideal is to create a software that automatically guides the user to the exact viewpoint. A second approach is to match the original and its rephotograph via post processing. This allows the photographer to create a well aligned picture pair, even though the exact viewpoint of the original is only approximated.

The automation of both processes requires a solution to a single underlying challenge. This is to automatically identify similarities between the historic image and its modern counterpart. Because of the use of different cameras, elimination changes and regular occlusions due to structural changes, this is a extremely challenging task for classic image matching approaches [Bae et al., 2010; Schindler and Dellaert, 2012].

1.2 Contribution

This work is tackling the challenge of automatically matching historic and modern images. Its main goal is to develop algorithms which detect similarities between images taken across long time periods. In practice, these similarities allow the post alignment of a rephotograph and its original. Furthermore, they can serve as a basis for automatic pose estimation to support rephotographers in the process of viewpoint recovery.

In this context the contributions of this thesis are manifold. At first, a small dataset is established, that consist of 52 rephotographs of scenes, which have been exposed to medium to tremendous change across the years. This dataset proves to be more challenging than popular datasets focusing on seasonal and day time changes. Thus, it may serve as a general challenge to assess the quality of newly developed feature detectors and descriptors. Furthermore, an online portal is developed that aims at collecting rephotographs from all over the world and currently contains more than 2000 image pairs. These may contribute to a larger dataset in the future.

Second, this work assesses the suitability of known features for aligning modern and historic image pairs, including classic detectors and descriptors as well as features extracted from modern pre-trained Convolutional Neural Networks (CNNs). It reveals that common detectors fail to identify features stable across the long time periods faced in rephotography. Instead, combined with dense sampling, individual binary and floating point descriptors as well as features extracted from pre-trained CNNs are applicable for rephotography alignment. Yet, all of them need to be integrated into a pipeline utilizing match filtering algorithms relying on geometry instead of photometry to distinguish between correct and false matches. The presented methods showed better results than all previous approaches.

Third, the first version of a mobile application that supports rephotographers in recovering the viewpoint of an image is presented, while its shortcomings and possibilities for enhancements are discussed.

In summary, this thesis presents fundamental investigations in computational rephotography, which in the future may serve to simplify the process of taking rephotographs. Besides, the developed approaches may once be suitable for automating tasks beyond rephotography, for instance the sorting of image archives [Schindler and Dellaert, 2012] or navigation in disaster zones, which experienced severe structural damages.

1.3 Structure

This thesis is organized as follows. Chapter 2 provides a brief introduction to the basic computer vision and photogrammetric concepts necessary to understand the following chapters. The history of rephotography from classical to computational rephotography and further works related to this thesis are reviewed in Chapter 3. Chapter 4 presents a detailed evaluation on the suitability of classic feature detectors, descriptors and match filtering approaches in the context of modern to historic image matching. A new method to measure the distribution of feature matches across an image is introduced in Chapter 5. Chapter 6 presents a follow up evaluation, which assesses the applicability of pre-trained CNN features for registering rephotographs as well as the possibilities to enhance alignment via semantic annotations. Chapter 7 focuses on practical developments, including the re.photos portal and different approaches to develop a mobile application that supports rephotography. Finally, Chapter 8 summarizes the main results of this thesis and provides ideas for future research in the context of rephotography.

Chapter 2

Technical Remarks

This thesis abstains from giving a detailed overview of the basic photogrammetric concepts underlying it as well as much of the related work. Instead, it is assumed that the reader is familiar with terms such as 6DoF camera pose estimation as well as Structure from Motion (SfM). To allow readers not familiar with these technical details to follow this work, this chapter gives a brief overview of the most important terms. Experts may simply skip this chapter. All other readers should be aware, that the following explanations are simplified and completeness is not guaranteed. For more comprehensive information the reader is expected to refer to established literature such as Hartley and Zisserman [2004].

2.1 Projective Transformation

A projective transformation is a 2D transformation that maps lines to lines but does not preserve parallelism, see Figure 2.1. With this a plane can be mapped to every other plane. Thus, if a chessboard is captured from two different views, these can be transformed into each other via a projective transformation.

2.1.1 The Homography Matrix

Every planar projective transformation can be described by an invertible $3\,x\,3$ matrix H manipulating homogeneous coordinates. Thus, every point x of the first plane can be mapped to the second plane via $x' = Hx$, while H is referred to as the homography matrix. A homography matrix has 8 degrees of freedom (DoF). Consequently, the homography relating two planes with each other can be computed as soon as four corresponding points on both planes have been identified, since each point correspondence provides two additional equations.

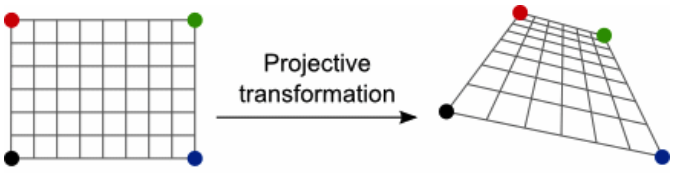


Figure 2.1: Illustration of a projective transformation[1].

[1] reprint from `https://www.graphicsmill.com/docs/gm/affine-and-projective-transformations.htm` (accessed on January 7th, 2020)

2.1.2 Automatic Corresponding Point Detection

Ideally the corresponding points of two images should be identified automatically. For this feature detectors, descriptors and matching algorithms are used.

Feature Detectors

Feature detectors identify stable points in an image. These are called keypoints and are supposed to be robust against rotation, scale and viewpoint as well as appearance changes. Consequently, keypoints are stable across images of the same object or scene and keypoints depicting similar elements of an object should be detected in all images of such. To detect stable keypoints several algorithms exist (see also Chapter 4), which apply different filters to the image, compute image gradients and identify image positions representing certain maxima. Finally, every keypoint features a pixel position on the image, a scale (radius) and an angle (rotation). However, depending on the algorithms applied, scale and angle might be fixed to default values.

Feature Descriptors

Feature descriptors describe the image area around a keypoint via a sequence of numbers. This either contains floating point values for floating point descriptors or is a combination of 0 and 1 for binary descriptors. The general idea is that for similar image regions a similar descriptor is computed independent of scale, perspective and illumination changes.

Again several algorithms exist that compute a descriptor based on an image patch (see also Chapter 4). In general, a keypoint's area, defined by its scale, is taken into account and the keypoint angle determines the direction in which pixels are considered. Then, diverse mathematical functions are applied to the pixel values of this area accumulating, filtering or weighing them. Finally, the output is a sequence of numbers featuring a constant length independent of keypoint scale.

Feature Matching

Feature matching uses feature descriptors to match the keypoints of two images. In general, direct nearest neighbour matching is applied to do so. This means, for each keypoint of the first image its descriptor is compared to every keypoint descriptor of the second image by computing the euclidean distance (L2-Norm) between both sequences. Finally, the keypoint with the closest descriptor is returned as a match. Two images with keypoints and matches between these are shown in Figure 2.2.

In Figure 2.2 only correct matches are displayed. In general the closer the appearance of the matched images the better the results of feature matching. Nonetheless, due to view and illumination changes as well as repeating elements in an image, false matches are common. These occur if two keypoints have close descriptors even though they do not represent the same object part in both images.

Match Filtering and Robust Homography Estimation

False matches need to be filtered out, either before or during homography estimation. Several filtering approaches applied directly after or during the course of matching, such as the ratio test [Lowe, 2004], are presented in Chapter 4. Furthermore, during homography estimation approaches such as RANSAC [Fischler and Bolles, 1981] can be applied to use only matches consistent with each other for model estimation.

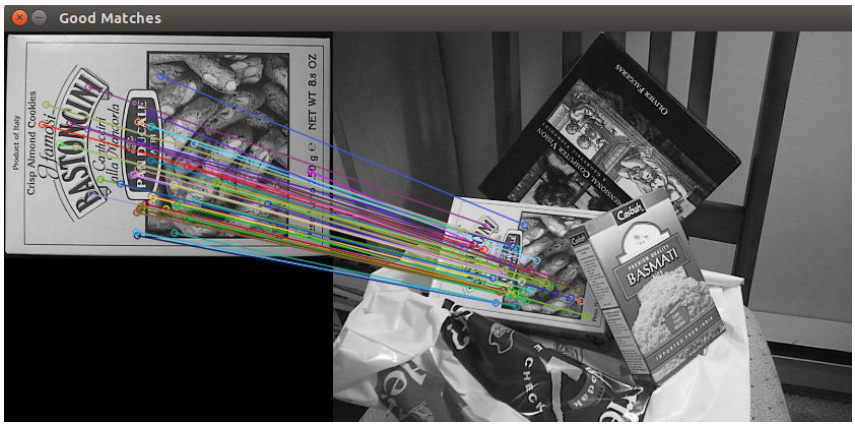


Figure 2.2: Illustration of a feature matching result[2].

2.2 6DoF Pose Estimation

Besides homography estimation, this thesis and related works aim at 6DoF pose estimation of an image. This means, provided a 3D model of the world, the 3D position of the camera as well as its rotation (additional 3 degrees of freedom) during image capture, should be recovered.

2.2.1 Camera Parameters

For projecting world points onto a camera frame and the other way around, a pinhole camera model is used, which makes several idealized assumptions. Among others this includes the assumption that the aperture size is infinitely small and all rays are projected through a single point, called the camera center, see Figure 2.3. With this assumption a camera matrix K can be established so that the projection of a 3D point X in camera-centric coordinates onto the image plane can be described by $x = KX$. Thus, given a 3D model of an object, this can be projected onto a camera image, whose camera center is located at the origin of the 3D world coordinate system.

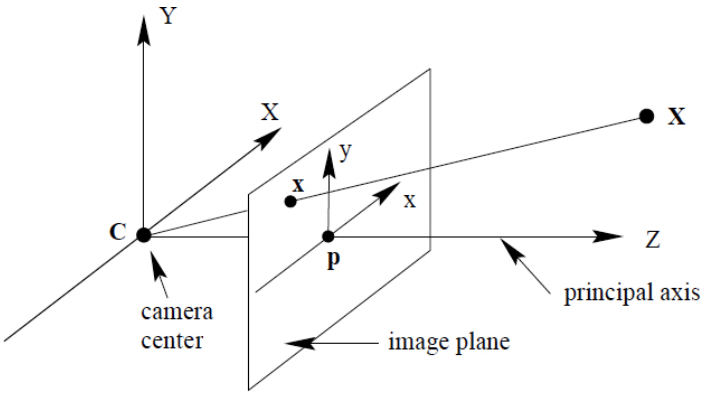


Figure 2.3: Illustration of a pinhole camera[3].

[2]reprint from `https://docs.opencv.org/3.4/d5/d6f/tutorial_feature_flann_matcher.html` (accessed on January 7th, 2020)

[3]reprint from `https://hedivision.github.io/Pinhole.html#ref1` (accessed on January 7th, 2020)

Extrinsic Camera Parameters

In practice the camera center is not always placed at the origin of the world coordinate system and often projections from more than a single camera need to be considered. To allow an arbitrary placement of the camera in the 3D world and still compute the projection of 3D objects onto the camera image, first the world coordinates have to be transformed to camera coordinates. This involves applying a $3\ x\ 3$ rotation matrix R as well as a $3\ x\ 1$ translation matrix T so that

$$X_{cam} = RX_{world} + T \tag{2.1}$$

Then the projection of X_{world} onto x on the image plane can be described by

$$x = K\,[R\,|\,T]\,X_{world} \tag{2.2}$$

Rotation matrix R and translation matrix T are called extrinsic camera parameters and feature 3 degrees of freedom each. Thus, for 6DoF pose recovery of an images viewpoint R and T need to be determined.

Intrinsic Camera Parameters

Next to extrinsic camera parameters there are also intrinsic camera parameters. These are integrated in K and describe the deviations from the pinhole camera model including the coordinates of the principal point, focal length in pixel dimensions and skew. These parameters can be determined for each camera via calibration with a calibration object, whose exact dimensions are known [Zhang, 2000]. However, intrinsic camera parameters are less relevant for this thesis. The reader should only keep in mind that camera intrinsics are usually unknown in rephotography, so that 3D reconstruction suffers from certain ambiguity.

2.2.2 3D Reconstruction

Previously the projection from a 3D model onto the camera frame was described. Now, as soon as more than a single view of a 3D object or scene is available the same formulas can be used to establish a 3D model.

Two View Geometry

In case two views of a scene are available, scene points visible in both images can be used to compute the essential matrix, which relates both views with each other. With this the 3D structure of the visible scene or object can be reconstructed up to a similarity transformation (undefined only in scale) in case both cameras are calibrated. Instead, with uncalibrated cameras the scene can only be reconstructed up to a projective transformation, see Figure 2.4. Yet, additional constraints, such as the orthogonality of sets of parallel lines, can be utilized to receive a 3D model only undefined in scale, even for uncalibrated cameras. Thus, in practice two views are sufficient to create a model approximating the 3D structure of a scene.

Structure from Motion (SfM)

Commonly more than two views of a scene are available, as in Structure from Motion (SfM), were a single camera moves around a scene and takes several images of this, see Figure 2.5. In this case scene points are visible in many frames. Thus, after computing the relative orientation between

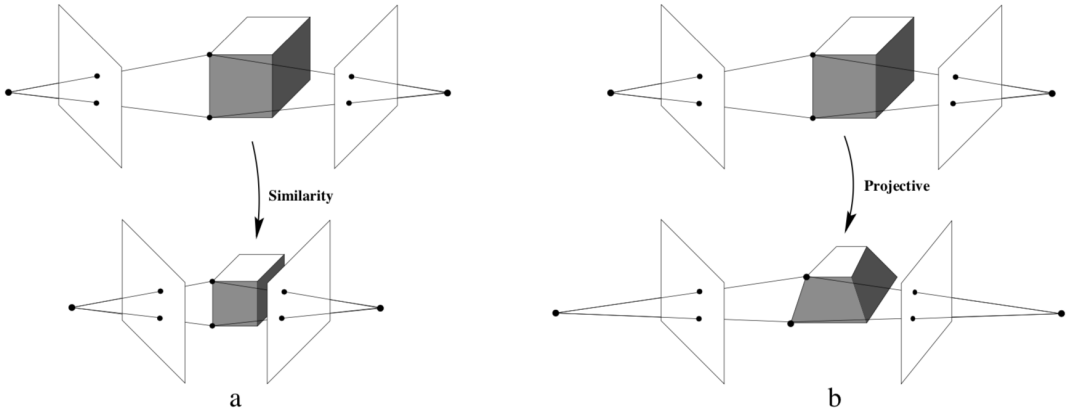


Figure 2.4: Illustration of 3D reconstruction ambiguity [Hartley and Zisserman, 2004].

two views and creating an initial 3D model of the scene, this can be refined with every new frame. This is achieved by creating a large system of non-linear equations and utilizing methods such as the Levenberg-Marquardt algorithm [Levenberg, 1944] to minimize the reprojection error of the points of the 3D model into each camera frame. In general, the utilization of more frames leads to a more accurate 3D model.

In rephotography SfM can be used to reconstruct the modern scene in 3D via several images from different view points. Afterwards, the position of the original historic image can be recovered by 6DoF pose estimation from the 3D model.

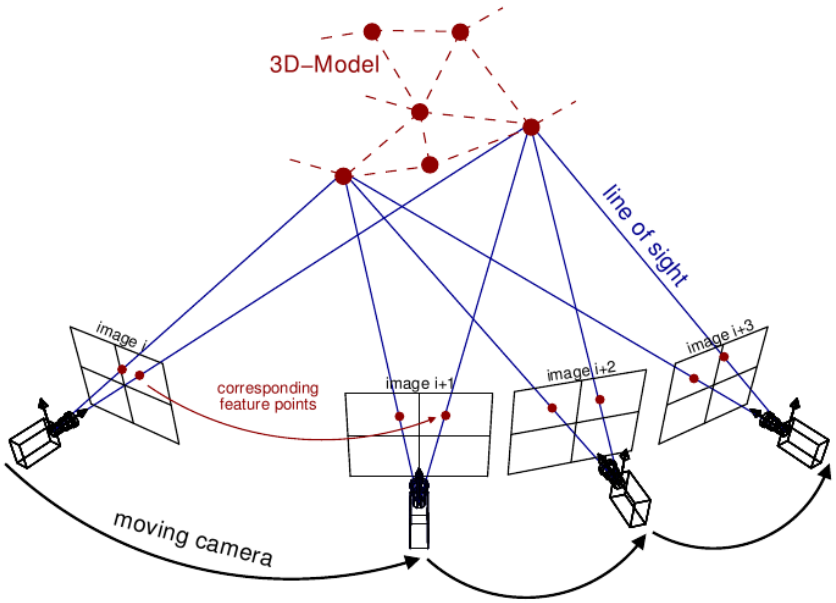


Figure 2.5: Illustration of the creation of a 3D model via Structure from Motion (SfM)[4].

[4]reprint from `http://theia-sfm.org/sfm.html` (accessed on January 7th, 2020)

Chapter 3

History and State of the Art

This chapter briefly outlines the history of rephotography. It starts with an overview of classical rephotography, as practiced for decades, including several methods to manually recover the vantage point of a historic image. These methods are based on the availability of different types of cameras ranging from the first film cameras, across instant to modern digital cameras. Besides, a short introduction to post-processing rephotographs with Photoshop is given. Afterwards, several mobile applications allowing computationally assisted viewpoint recovery as well as more scientific approaches to the problem are presented, including a method that creates artificial rephotographs of a scene. Finally, related work beyond the scope of rephotography is summarized.

In this chapter rather general related work is mentioned. For research related to specific parts of this thesis, please refer to the corresponding chapters.

3.1 Rephotography

Rephotography, also referred to as repeat photography, has been practiced for more than one century. As early as 1888, Sebastian Finsterwald documented the movement of mountain glaciers in the Tyrolean Alps, by repeatedly photographing them across two years [Hattersly-Smith, 1966]. In turn rephotography has been mentioned as a research method in ecology by Clements [1905].

Since then, several photographers have used rephotography to provide compelling visualizations of historic change. Popular works include "New York Changing" by Levere et al. [2004] and the "Then and Now" books of Thunder Bay Press, of which each accumulates rephotographs of a major world city [e.g. Campi, 2008; Reiss and Joseph, 2013]. While these works document urban changes others primarily focus on rural areas. Examples are "Second View" [Klett et al., 1984] and "Third View" [Klett, 2004], which emerged from two rephotographic survey projects conducted in the American West.

Next to publications addressed at the interested citizen, several more exist, that have a primarily scientific purpose. In ecology, rephotography is widely-used to document changes in vegetation and climate [Clements, 1905; Hastings and Turner, 1965; Rhemtulla et al., 2002]. Other works include the documentation of glacier melting [Gore, 2006; usgs.gov] and the monitoring of geological erosion [Hall, 2002]. Moseley [2006] and Salick et al. [2005] studied the impact tourists and locals have on the Himalayan ecosystem. Sociologists on the other hand use rephotography to study urban development and social change. For instance Walker and Leib [2002] documented the influence migrant workers have upon their temporary communities.

Robert H. Webb, a USGS hydrologist, who replicated 445 pictures of the Grand Canyon originally taken by Robert B. Stanton in 1889 and 1890, mentions:

> "There is no more basic scientific technique than interpreting old photographs. We discovered changes in the Grand Canyon that could not have been determined any other way." [Webb, 1996, pp. 214]

Thus an interest in rephotography exists for the general population as well as for scientific purposes, especially, since the visual medium picture is easily understandable and often particularly meaningful.

3.1.1 Classical Rephotography

The goal of each rephotographer is to duplicate the original photograph as close as possible. This includes retaking the photo from the original vantage point and sometimes even replicating the original lightning conditions, considering daytime as well as season. A further challenge is the recreation of the dimensions of the objective lens.

At first, it is necessary to acquire the original image itself. This is usually acquired from archives or museums. In the past, the photographer often owned a glossy print of the original, but sometimes he first needed to duplicate or copy negatives. Nowadays, many museums have digitized their archives and some additionally made them publicly available online. As a consequence acquiring images has already become more easy. However, often pictures are only accompanied by a broad description of the location they were taken at. So the photographer is often left with the content of the picture. In the ideal case a famous building is shown, while other times it is only known, that the picture belonged to an archive from the grand canyon. In the first case the location is known up to a few meters, while in the second a great terrain needs to be explored. This is usually realized by asking people, who know the region very well, or by browsing through similar archives with more specific location information.

However, even knowing the location up to a few meters the photographer still needs to locate the exact vantage point. Most rephotographers do this by eye, see Figure 3.1. They take turns in looking through their lens and at the picture, move the camera to the right, left, forward, backward, upward and downward and rotate it until they find the view most close to the original. To determine the height of the picture it helps to know the original photographers height approximately or the camera and tripod he used. Knowing the camera that was used is also helpful to determine the zoom and curvature of the original camera lens, only if this is duplicated as well an exact rephotograph can be taken. For this reason an additional picture of the original photographer posing with his camera can be very helpful. Yet, often this additional information is unavailable. In this case a rephotographer needs to identify the difference between parallax and zoom to approximate the original picture.

In the following a detailed description of the strategies used by professional rephotographers such as Malde [1973] and Klett et al. [1984] is given. Besides, a more recent approach utilizing the grid overlaid on the electronic viewfinder of modern cameras is presented, followed by different strategies for post-processing rephotographs.

Malde [1973]

In 1973, Malde described a strategy for determining a photographs exact location utilizing the principles of parallax. This technique requires several distinctive and persistent details equally spread across foreground and background of the picture. Provided these conditions, the idea is

Figure 3.1: Illustration of the process of capturing a rephotograph by eye. Image from the Asa Kinney Project blog by James Gehrt [2014].

to first establish a vertical line on the original photograph as shown in Figure 3.2. Now, this imaginary line is located in the scenery by aligning features along both sides of the line as in the original. Second, the focus is on objects in the left and right part of the image further away from the center line. As an alternative to single objects, orientation lines connecting two landmarks on each side of the center line may be used, see also Figure 3.3. Moving forward and backward along the line these are correctly arranged, so that the spot the original picture was taken from is determined. Finally, the height of the camera can be determined via comparing the position of foreground and background artifacts on each side. Figure 3.2 shows an example of a very accurate replication, while the reference lines in Figure 3.3 suggest that the rephotographer stood several meters too far to the left of the original vantage point.

Harrison [1974] and Klett et al. [1984]

A more accurate and mathematical method to discover the original vantage point has been presented by Harrison [1974]. This technique was applied by Klett and others taking the pictures for their Second and Third View Survey Projects [Klett, 2004; Klett et al., 1984] and is clearly presented in a tutorial on the Third View Project website [thirdview.org].

Harrison notes that there are six degrees of freedom determining the exact position of the original picture. The first three define the camera position in three dimensional space, while the others describe the rotation of the three orthogonal axes of the image plane. In theory, provided six independent measurements these unknown values can be reconstructed. Since Harrison measures the distance between identifiable points, at least five similar points need to be identified in the original as well as an instant print out of the current scene.

First of all the distance between two points in the far back should be measured, see Figure 3.4a. Points in the far distance remain stable among slight position adjustments. Thus, their distance can be used to normalize the following measurements between the original and the new instant print outs, as their scales usually differ, see Figure 3.4b.

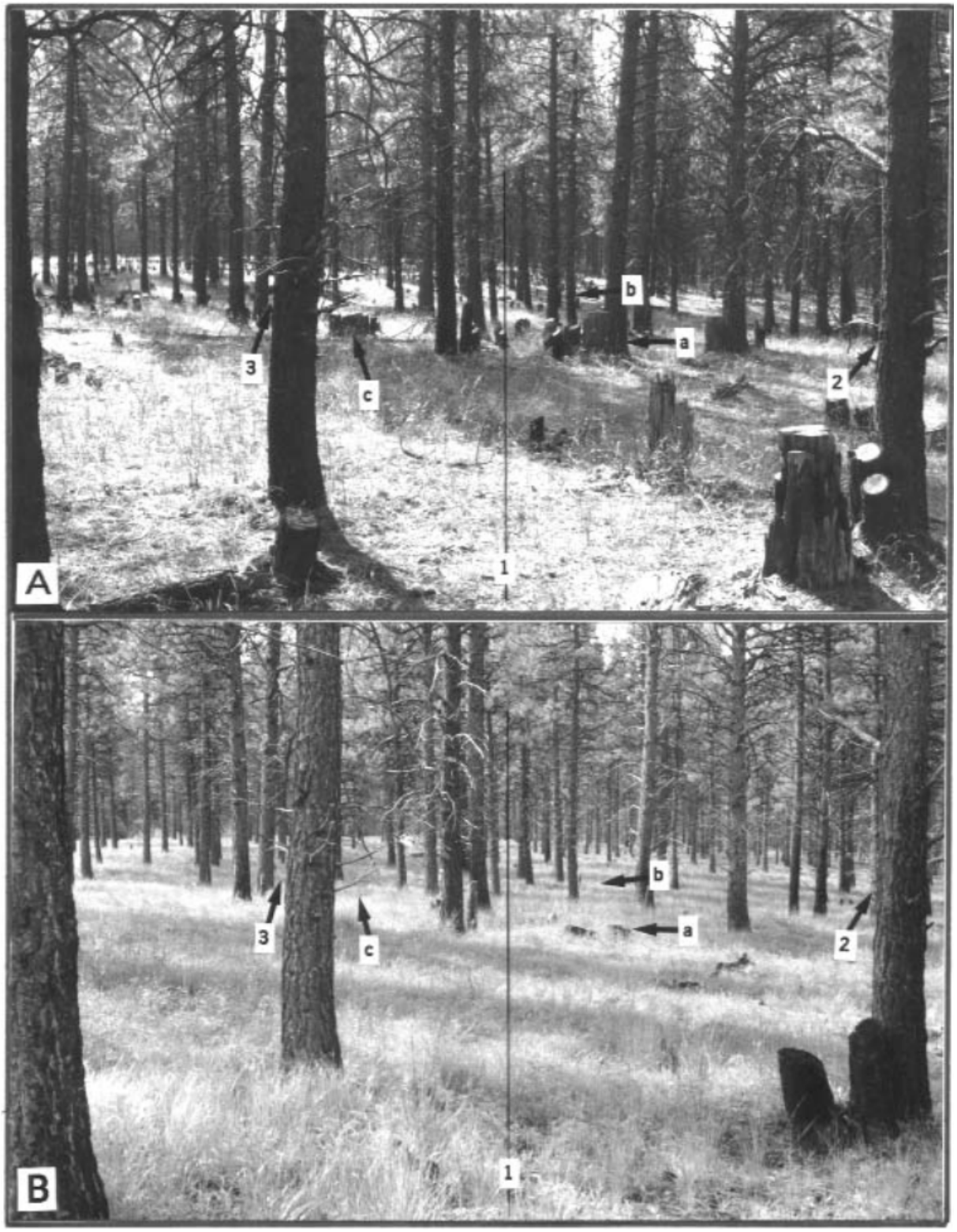


Figure 3.2: Illustration of the method proposed by [Malde, 1973]. At first, vertical line (1) is established in the original (A) and the current view (B) is brought into alignment by the position of trees close to the line. Second, details (2) and (3) are used to determine the correct positioning along the imaginary line. Finally, by comparing more features in both background and foreground the height of the camera is adjusted. [Hall, 2002]

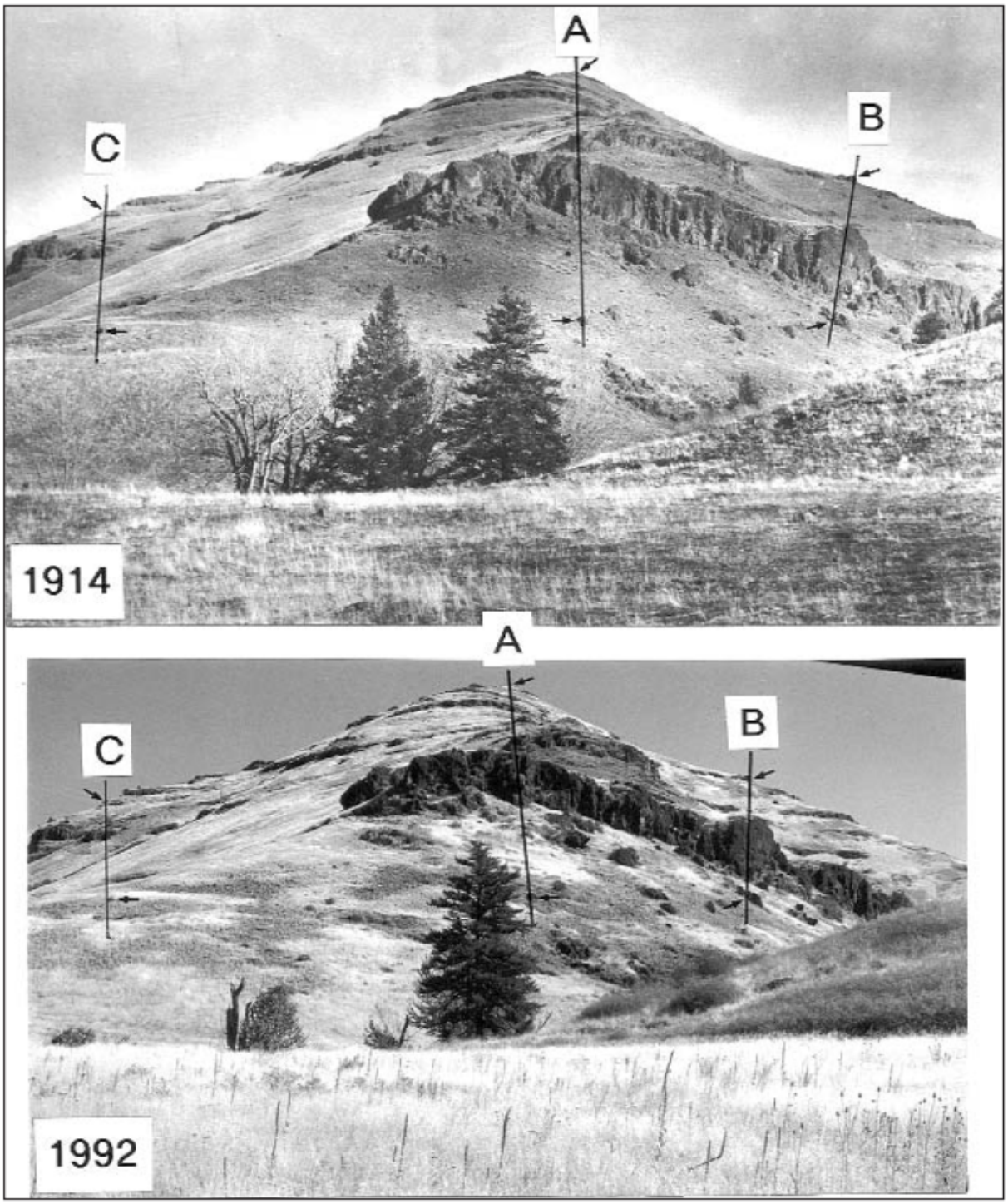
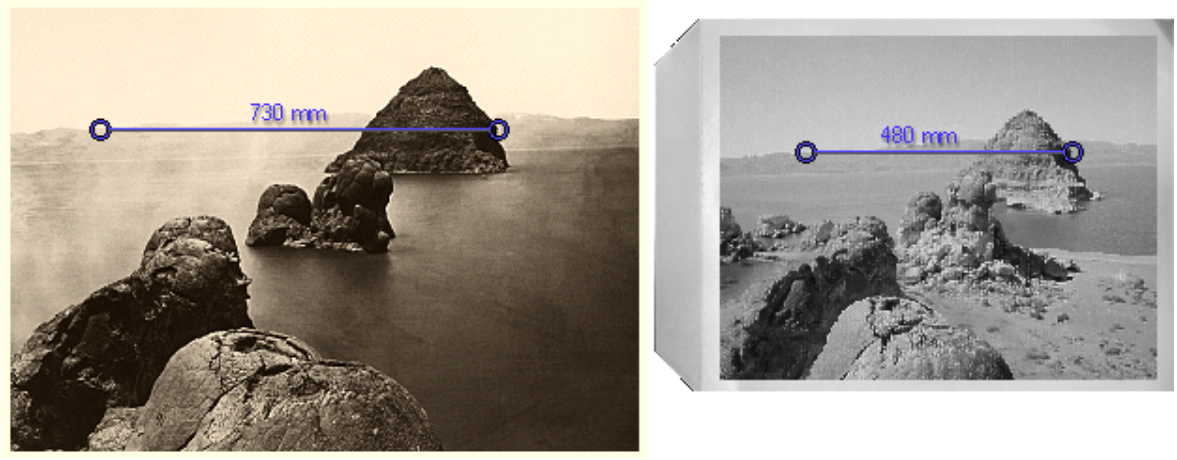
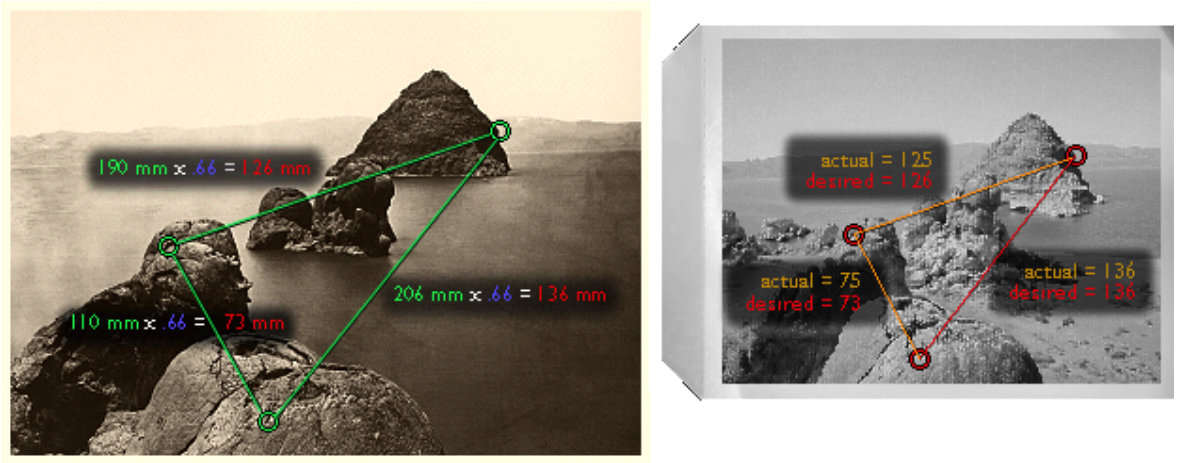


Figure 3.3: Illustration of the method proposed by [Malde, 1973] on a photo of Branson Creek, Wallowa County, Oregon taken in 1914. The rephotograph (bottom) was taken by Skovlin and Thomas [1995]. Here (A) is the vertical line first established in the original, while (B) and (C) are orientation lines further away from the imaginary line used for correct positioning. Orientation lines instead of single landmarks allow a more precise relocation especially in wide territory. The difference in the angle of all three orientation lines suggests that Skovlin and Thomas [1995] positioned themselves too far to the left. [Hall, 2002]



(a) First the distance between two points at the far back is measured to calculate the scale difference between both photos.



(b) Second, further measurements are taken and normalized by the calculated scale difference. The vantage point has been successfully reconstructed as soon as all normalized measurements agree.

Figure 3.4: Illustration of the method proposed by Harrison [1974] and Klett et al. [1984]. Image pairs from the tutorial at thirdview.org. Original (left pictures): Timothy O'Sullivan, 1867. Rock Formations, Pyramid Lake, NV.

As soon as all normalized measurements are the same the original vantage point has been found. However, till this state is reached some patience is required. Klett et al. [1984] nicely state:

> "Moving up increases distance between foreground and background"
> "Moving right shifts objects to the left"
> "Moving forward pushes objects out from the center" [thirdview.org]

Yet, movement into a single direction usually influences more than one measurement. Consequently, a distance value that was very close to the original at first, may drift further away again as the photographer tries to align another measurement. As a result some experience is required.

Additionally, Harrison [1974] recommends to construct axes perpendicular to one of the distance measurements. For an example see Figure 3.6. The ratios of the line segments $E'C'$,

Figure 3.5: Illustration how to utilize the electronic viewfinder of modern cameras to relocate the original vantage point of an image. Left: Photograph by T.J. Hileman from the early 1900s. A grid structure identical to that of the viewfinder of the modern camera has been drawn onto the picture. Right: Look through the modern cameras viewfinder exhibiting the same grid structure. [Smith, 2007]

$C'D'$ and $D'E$ of EE' allow to compare the parallax of different objects between the original and the new photos. Ideally, the distance measurement used for construction is approximately parallel to the image horizon, as the measurement in Figure 3.4a. In this case, the measurements of the line segments best fit the three possible moving directions.

Electronic Viewfinder Strategy

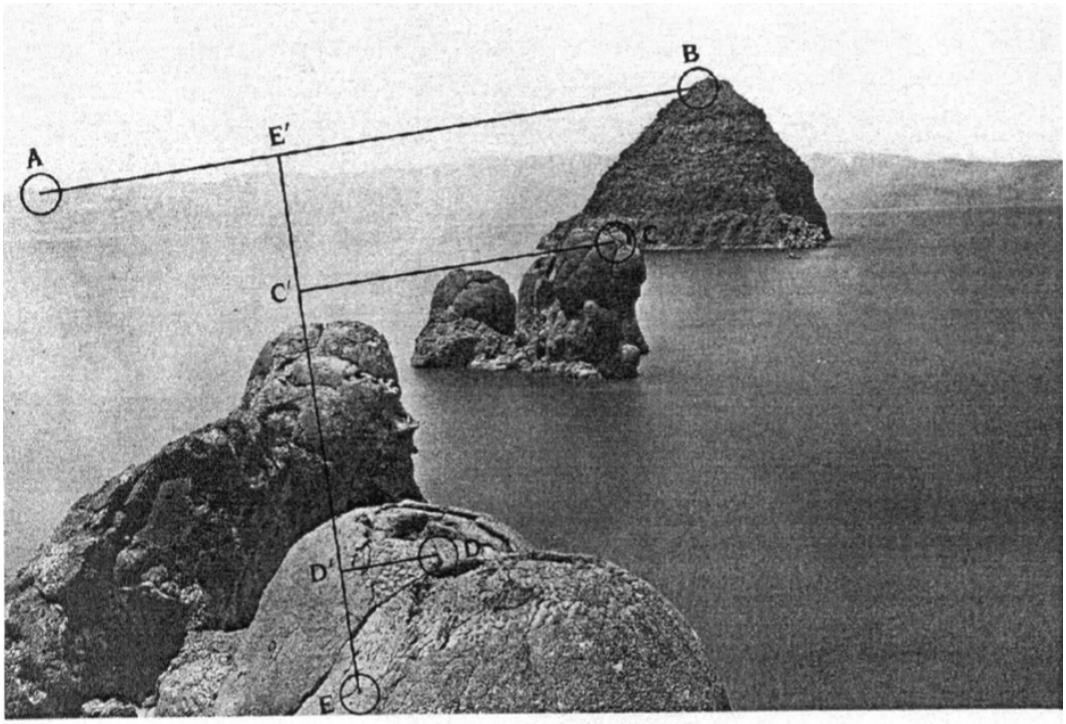
With the emergence of digital cameras in professional photography at the end of the 1980s a third strategy for viewpoint reconstruction developed. This utilizes the grid these cameras overlay on the image of the electronic viewfinder.

The idea is to draw a replication of the grid structure of the modern camera onto a print out of the original photograph, see Figure 3.5. The additional lines and intersections allow the comparison of the placement of landmarks in the original image and the current camera view. Similar to the methods of Harrison [1974] and Klett et al. [1984], with a little practice the displacement of landmarks depicts the necessary direction of movement to the photographer. For instance, if features in the background are to far above a horizontal line and features in the front are to far below it, the photographer is to close to the scenery and needs to move backward to reach the original vantage point. The magnitude of derivation together with the knowledge of approximate distance of the considered landmark, in other words whether such is part of the background or foreground, provides information on the magnitude of movement necessary. This way one step after the other the original vantage point is reconstructed.

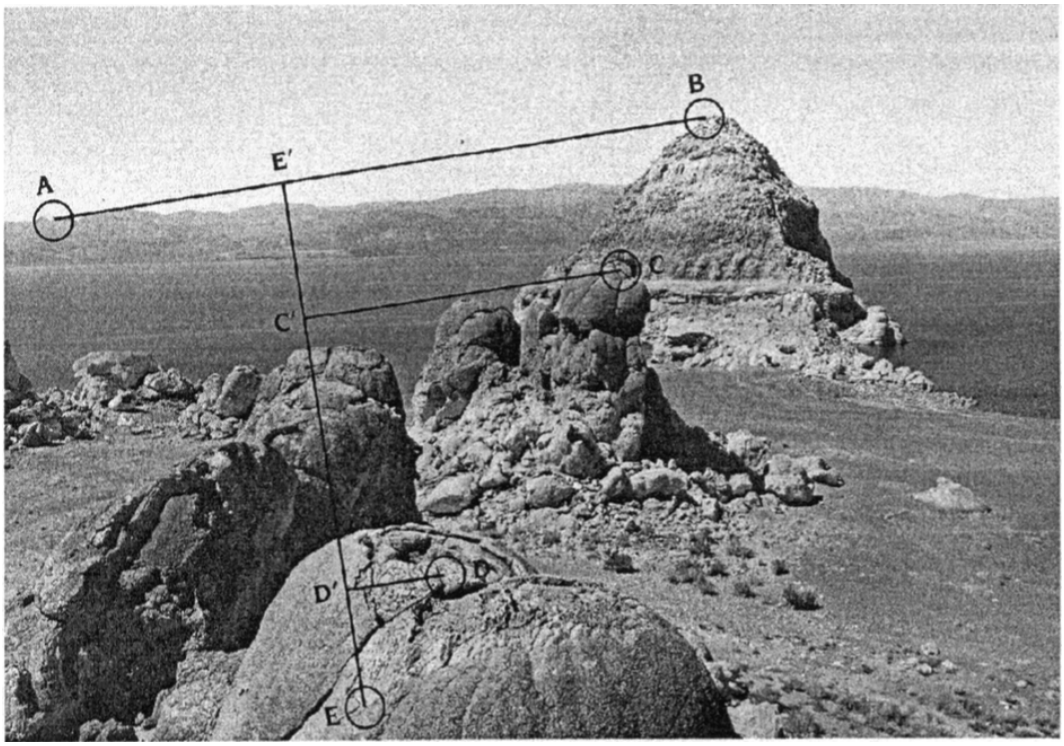
Discussion

Even though the three methods mentioned above provide clear instructions they are time consuming. The strategy of Malde [1973] is rather vague and demands several persistent features equally spread across the foreground and background of the picture. The method of Harrison [1974] on the other hand is more precise but requires lots of equipment, as Klett [2004] state:

> "Our essential tools are a copy of the picture to be rephotographed, a view camera (a large camera that permits perspective and distortion control), a tripod, instant Polaroid film, a ruler in .5mm increments, a fine-point pen, and a calculator." [third-view.org]



(a) Timothy O'Sullivan, 1867. Rock formations. Pyramid Lake, Nev. (Massachusetts Institute of Technology)



(b) Mark Klett for the Rephotographic Survey Project, 1979. Pyramid Isle. Pyramid Lake, Nev.

Figure 3.6: Illustration of the method proposed by Harrison [1974] and Klett et al. [1984]. EE' is constructed perpendicular to the distance measure of AB and afterwards CC' and DD' are drawn perpendicular to EE'. As a result the ratios of the line segments $E'C'$, $C'D'$ and $D'E$ of EE' allow to compare the parallax of different objects between the original and the new photos.

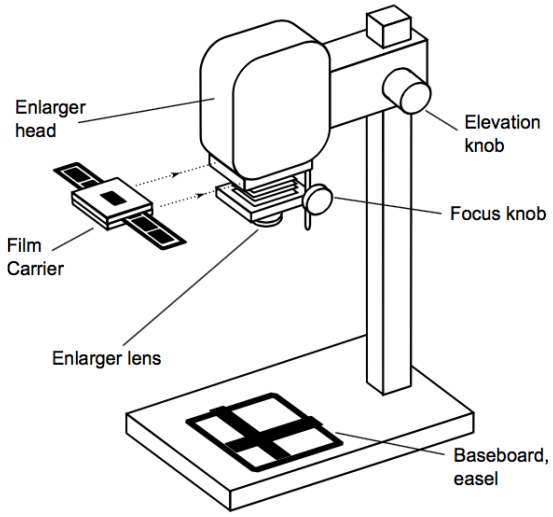


Figure 3.7: Scheme of a photographic enlarger[1].

To get an idea of the necessary work we simply need to keep in mind that all measurements are made on instant print outs and need to be repeated every time the cameras position is adjusted. Furthermore, the methods of Malde [1973] and Harrison [1974] both assume that the interior camera parameters have already been successfully reconstructed. This means the modern cameras lens has a similar zoom and curvature as the camera that took the original image. Otherwise, the taken measurements will never align altogether.

The last method presented is less time consuming, since it does not require instant print outs but provides the user with measurements directly visible in the scenery via the viewfinder. Yet, even with this method measurements regularly need to be compared to those of the original print out. Consequently, there is lots of room for improvement. Further approaches utilizing computer and mobile devices to support vantage point reconstruction in rephotography are presented in Section 3.1.2.

Post-Processing

After recapture, post-processing is applied to improve the alignment of the modern photo and the original. In order to do so the new picture is resized, cropped and rotated. In the past, the tool utilized for this was a traditional photographic enlarger as shown in Figure 3.7. This enlarger allowed to produce photographic prints from negatives and manipulate their size via the integrated lens.

Nowadays, post-processing is done electronically by photographic software that is much more flexible. This allows to change scale and rotation of the image, but also to manipulate curvature. Furthermore, digital post-processing makes it possible to split the image into different segments and apply different image transformations to each. This enables subsequent correction of the alignment of all image areas. Before, in case the deviation between the original and new photograph was too strong, the photographer needed to focus on a single object or area. However, if this object in the foreground was brought into perfect alignment, the image parts on the side and in the background often no longer matched well.

[1] reprint from `https://commons.wikimedia.org/wiki/File%3ADarkroom_enlarger_en.svg` (accessed on January 7th, 2020)

Figure 3.8: Illustration of the power of the *Photoshop Warp Tool*. From left to right the diamonds are dragged to align with the red structure wrapping the tree[2].

To the author's knowledge there is no research paper elaborating the possibilities of digital post-processing in the context of rephotography. Hence, in the following this work resorts to the digital editing process described by some rephotography blogs[3] [Gehrt, 2014]. The tool of choice used for post alignment seems to be *Photoshop*, specifically *Photoshop's Warp Tool*. This allows to manipulate the shape of an entire image or single image patches by simple drag and drop operations. In practice, a 3x3 grid is projected onto the area of the image that is supposed to be distorted. This grid can be dragged at all points or lines inside of it allowing to change its entire shape as illustrated in Figure 3.8.

In his blog James Gehrt describes how to use this technique for rephotographic alignment [Gehrt, 2014]. First of all, he places both images on top of each other and adjusts the transparency of the modern image to make the original show through. After adjusting the scale and perspective of the rephotograph he applies the *Warp* method to manipulate single parts of the image, see Figure 3.9. He notes, that during the process it is important to pay attention that straight lines and buildings remain straight. Overall, he goes back to adjusting scale and perspective several times before reaching the final result. Hence, he states that the whole process is often tedious and time consuming.

In general, digital post-processing is a big progress in the context of rephotographic alignment. Yet, as the possibilities of subsequent manipulation have increased and these all need to be initiated manually the process is not less time consuming.

3.1.2 Computationally Assisted Rephotography

With the emergence of affordable compact but high quality digital cameras as well as mobile devices, that take qualitative photos and are always at hand, acquiring rephotographs is no longer a topic limited to a minority of professional photographers and scientists. Nowadays the general public is interested and involved in the process of creating rephotographic pairs as the availability of mobile applications in this domain shows.

In the following first several publicly available mobile applications are presented. All of these support general users in acquiring rephotographs by utilizing the digital possibilities these devices offer. Afterwards, four different computationally driven scientific approaches for creating rephotographs are presented. These include a software prototype that supports a user in recap-

[2]Images from the tutorial of Dennis Dunbar at `https://design.tutsplus.com/tutorials/making-sense-of-the-warp-tool-its-all-about-the-lines--psd-3896` (accessed on January 7th, 2020)

[3]Re-Photography: Practitioners, 2014 `http://stelioshadjielias100.blogspot.de/2014/05/re-photography-practitioners-markklett.html` (accessed on January 7th, 2020)

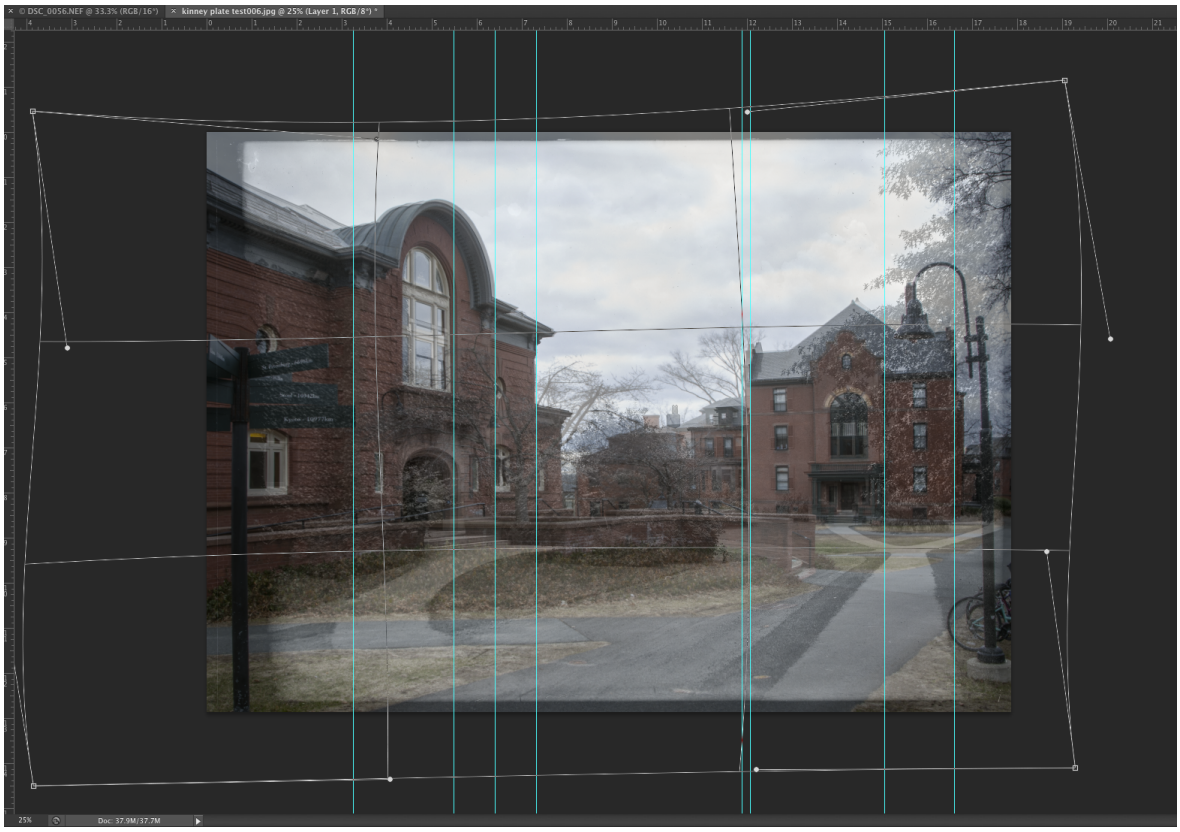


Figure 3.9: Illustration of James Gehrt's approach utilizing *Photoshop* for the post-processing of rephotographic images. [Gehrt, 2014]

turing the original vantage point of a picture, as well as three methods that create rephotographs from image collections. The latter do not necessarily require someone to go out into the field.

Mobile Applications

Timera[4] combines a website with a mobile application for Android as well as iOS. The website features several historic images that can be uploaded by users and are saved with their corresponding GPS location. The mobile application allows to select an original from this collection or from ones own portfolio. It assists in retaking the respective picture by superimposing the camera preview with the original image. Additionally, the original image can be zoomed to better fit the live camera view. After recapture the user needs to define the final superimposition of both images. This is, as on *Timera* not two images are presented side by side, but a single image is composed of a rephotographic pair, as shown in Figure 3.10. As a result the user needs to define the area of the original image which should superimpose the new image as well as the grade of transparency applied. Consequently, the user has the possibility to completely cover one part of the new image or let it shine trough. This way minor perspective differences may be suppressed.

rePhoto[5] is another rephotography website accompanied by a mobile application. It is a joint project of Washington University, University of Vermont and the University of California, San

[4] `http://www.timera.com` (accessed on April 30th, 2017)
[5] `http://www.projectrephoto.com` (accessed on January 7th, 2020)

Figure 3.10: Rephotograph from *Timera*[6]: Kure Beach, United States by Jo Rockstar.

Diego, partially supported by the National Science Foundation. Similar to *Timera* the mobile application features a map with nearby spots to recapture. Furthermore, it supports its users during image capture by different overlays of the original onto the live camera view. The user can choose between a transparent overlay, a complete superimposition on the left, right, top or bottom half of the screen and several vertical or horizontal stripes. Even though the application allows private rephotography as well it originally focuses on a scientific purpose. Originally it was designed to involve the public in monitoring environmental changes, such as the growth of trees. For this reason, on the website no single rephotographic pairs are presented. Instead, the site is organized into different projects covering larger areas featuring several photo locations, while photos belonging to a single location are presented on a time line.

RePrism[7] is another standalone mobile application for rephotography. As the previous applications it supports taking rephotographs via a transparent overlay onto the life camera view. Apart from this no additional functionality is provided.

Computational Rephotography [Bae et al., 2010]

Bae et al. [2010] present a software prototype which employs modern computer vision techniques to guide a rephotographer to the viewpoint of a historical photograph. Their approach can be classified as semi-automatic, since in the beginning it requires manual user interaction to identify similarities between the current view and the historical image as well as for reconstructing the intrinsic parameters (e.g. focal length, principal point) of the original camera. Thereafter the computation of the required movement direction is completely automatic and the user simply needs to follow the provided navigation instructions.

Bae et al.'s [2010] preliminary studies revealed that manual rephotography is extremely challenging especially for untrained users. They report that neither with a side-by-side visualization, as in classical rephotography, nor with a linear blend of the original and the current

[6] `http://www.timera.com/t/68ugaEmo` (accessed on April 30th, 2017)

[7] `https://play.google.com/store/apps/details?id=com.online_siesta.reprism` (accessed on April 30th, 2017)

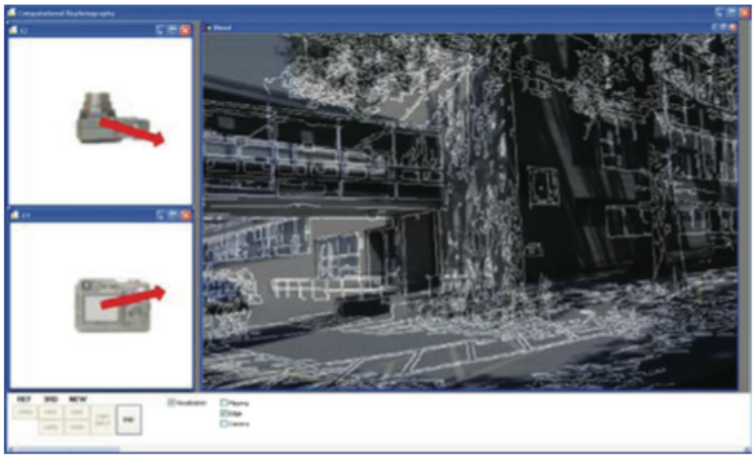


Figure 3.11: Illustration of Bae et al.'s [2010] technique to display the required movement to the user.

camera view, as in the previously presented applications, subjects succeed in reconstructing the vantage point of the original image. In summary, among others Bae et al. [2010] identify the following challenges for computationally assisted rephotography, which they solve with a combination of algorithms and manual user interaction.

1. The communication of the necessary movement to the user, which involves 3D translation as well as rotation.
2. The degeneracy of pose estimation as the user approaches the original vantage point due to little motion between the current and original view.
3. The identification of similarities between historical images and modern photographs.
4. The unknown intrinsic calibration of the original camera.

At first, their software approximates the intrinsic and extrinsic parameters (the 6DoF pose) of the original camera. This is achieved by creating a sparse 3D reconstruction of the modern scene from two images captured by the user, one taken close to the original vantage point and the other approximately 20 degrees away from it. Afterwards, the user is asked to identify the location of 6 to 8 reconstructed 3D points in the original, so that its 6DoF pose relative to the captured images can be computed. This way Bae et al. [2010] avoid the need to automatically identify similarities between the original photograph and the modern scene as mentioned in challenge (3). Furthermore, the user is asked to identify parallel lines in the original image, to allow principal point estimation via vanishing point identification, which helps to deal with challenge (4). In the following only the relative pose between the current camera view and the previously captured distant image (20 degree away) is computed to derive the necessary movement. For this a well-established feature detector and descriptor such as Scale-Invariant Feature Transform (SIFT) [Lowe, 2004] can be used and the degeneracy of pose estimation mentioned in (2) is avoided.

Finally, Bae et al. [2010] decided to only present the required translation to the users. This is displayed via two 2D arrows, one showing the need to adjust the cameras height and the other displaying the direction the user should move to, see Figure 3.11. The rotation is automatically corrected via warping the current camera view with a homographic transformation, which can also adjust camera zoom. In experiments this solution to challenge (1) proved to be more efficient than a 3D camera pyramid visualization able to simultaneously display the required translation and rotation.

Bae et al. [2010] conducted a user study to compare their approach to simple linear blending. In this their approach proved to be more efficient for recapturing the original vantage point of an image. While the average deviation from the original vantage point was 4.4m (standard deviation: 2.9m) using linear blending, their advanced method resulted in an average deviation of only 1.8m (standard deviation: 0.6m). Yet, they also report on several shortcomings. As users get closer to the original viewpoint they adhere more to the alignment blend, since pose estimation often fails to account for small movements. Furthermore, the captured scene needs to feature enough texture and 3D structure to allow accurate pose estimation. In other words, first of all enough stable feature points need to be identifiable in the scene (around 20) and secondly these need to be spread across the entire image and belong to different image planes. Instead, if all feature points lie roughly on a plane, for instance the front of a single building, every view can be registered by a homography which prohibits pose estimation. Additionally, Bae et al. [2010] noticed that "users typically focus on landmarks in the center of the image, and may not notice that features towards the periphery are not well-aligned."

In summary, Bae et al. [2010] presented a promising approach. Their ideas clearly contribute to simplify the rephotographic process. Unfortunately, their work can only be considered as a proof of concept, as their prototype has not been further developed nor publicly released. Presumably, this is also due to the shortcomings mentioned.

Rephotography via Image Based Rendering [Lee et al., 2011]

Lee et al. [2011] present an entirely different approach to create a rephotograph that does not require the photographer to reach the original vantage point. Instead, the rephotograph is constructed from a collection of images. To achieve this the user needs to capture several views of the scene shown in the original image. These are used to construct a 3D point cloud of the current scenery. Now, similar as in Bae et al. [2010], the user is required to identify similarities between the point cloud and the original image. So the relative pose of the original in reference to the point cloud and all other images can be computed. Finally, image rendering techniques Shum and Kang [2000] are used to render the rephotograph from the image collection.

An example rephotograph is displayed in Figure 3.12. Compared to a rephotograph taken by an actual camera, the result is composed of numerous blurred edges and shows several unaesthetic artifacts. The edges of the roof are not continuous and the image shows holes, red spots in the lower left corner, were pixels could not be inferred from the image collection. In the end, the presented approach relies on an image collection accurately composed of different view points. Yet, even given such the resulting photograph will never be as sharp as one captured by an actual camera and not please many photographers.

Internet Rephotography [Shrivastava et al., 2011]

Shrivastava et al. [2011] propose the idea of internet rephotography. They developed a method for cross domain image matching including paintings, sketches and photos of diverse ages. In detail they use Histogram of Oriented Gradients (HOG) as features to describe their images. The weights of these features are learned via a data-driven uniqueness assumption. This supposes that an image class is best depicted by features hardly present in images of other classes. For training a linear Support Vector Machine (SVM) framework is used, which requires only a single positive example against a large background dataset. This way Shrivastava et al. [2011] succeed in extracting important image regions even if the images are from diverse domains.

Based on these features they generate a similarity score for a given image pair. This allows

Figure 3.12: Illustration of the results of Lee et al.'s [2011] rephotography technique. Top left: Original, Top right: Image Collection, Bottom: Rendered Rephotograph

them to find modern images similar to an historical photograph online. Among the five best matches the user can select an image to create a rephotographic pair. If both images match well only little manual alignment is required. Two examples are shown in Figure 3.13.

They named this approach Internet Rephotography. Internet rephotography provides sharp results and simultaneously requires very little user effort. However, it relies on the online availability of a huge amount of images of the original scene. This may be available for popular sights in major cities. However, less frequented regions still require the travel of individual photographers.

Painting-to-3D Model Alignment [Aubry et al., 2014]

Aubry et al. [2014] developed a method to align images from multiple domains (drawings, paintings and historical photographs) to a 3D model. Due to specific rendering style, age and lightning changes, these images appearance is often very different from that of the 3D model. To handle these appearance changes they represent each scene of the 3D model by a set of discriminative visual elements based on the HOG descriptor. These elements as well as the weights of individual features for each element are learned utilizing machine learning techniques. Finally, Aubry et al. [2014] show that they are able to align new images to the 3D model utilizing the previ-

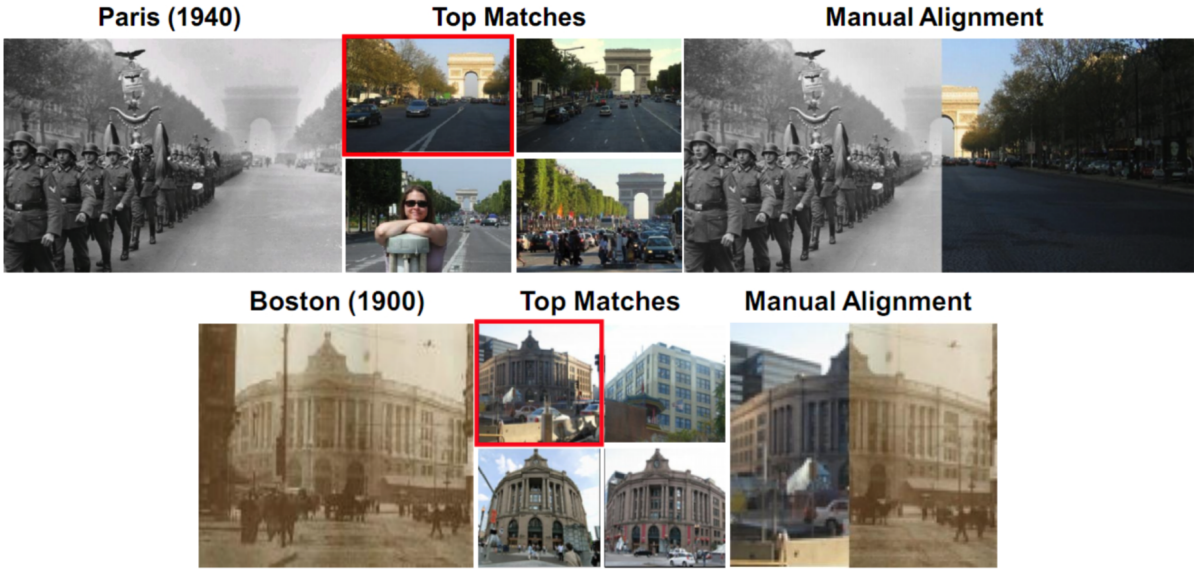


Figure 3.13: Internet Rephotography. [Shrivastava et al., 2011]

ously learned visual elements. Furthermore, they propose to use their method to automatically recover the original viewpoint of a historical photograph and recapture it within the 3D model.

Depending on the density of the 3D model such pictures may suffer from similar flaws as those of Lee et al. [2011]. Besides, a large 3D model of the respective location is required. Unfortunately, similar to large image collections, these are not available for many scenes.

Discussion

The presented mobile applications show that there is a general interest in rephotography. However, these applications provide only little assistance in retaking a photograph compared to the approach of Bae et al. [2010]. Consequently, a remaining challenge is to make more advanced assistance techniques for taking rephotographs available to the public.

The first three of the presented scientific approaches have in common that they require the user to identify similarities between the original historic image and the modern scene. Bae et al. [2010] and Lee et al. [2011] demand this explicitly at some point of their procedure. Even Shrivastava et al. [2011] who preselect the rephotograph candidate images automatically, clearly utilizing some similarity criterion, rely on the user to make the final decision on the modern photograph and to perform accurate alignment. Only the similarity criteria used by Aubry et al. [2014] are learned from a large 3D model of the respective site. Thus, another major challenge is to automatically identify similarities between historical and modern photographs, without the need to utilize large image collections. This could lead to further automation and therefore simplify the presented approaches.

All of the above approaches focus on the process of acquiring a rephotograph, while none concentrates on post-processing. This is surprising, since advanced post-processing is able to correct many flaws of imprecise view reconstruction and is commonly applied by professional rephotographers. In fact, *Timera* allows its users to perform some manual post-processing. Yet, there way of image presentation is very different from traditional rephotography and more or less eliminates the need for more advanced post-processing. Bae et al. [2010] also perform some post-processing as they correct the rotation and zoom of the live view automatically via a homographic transformation. However, even though they state that their rephotography results

highly depend on the users tolerance for error and often landmarks in the center are more well aligned than those on the periphery, they conduct no further post-processing. Nevertheless, as Section 3.1.1 described there are a variety of opportunities for subsequent image alignment. Therefore another challenge is to further automate post-processing.

The Potential Future of Rephotography

This paragraph gives a short overview of the future of rephotography, in which modern digital images will be recaptured. These are usually accompanied by additional information, which is able to ease the rephotographic process. Nonetheless, there is a persistent need to improve present rephotographic procedures, which is justified at the end.

Modern photos usually exist in high quality electronic format. Apart from the image itself this electronic format is composed of so called Exif data that provides lots of additional information. This includes an identifier of the device that took the image as well as other useful information such as the focal length or exposure time. Additionally, the date and time of capture are recorded and in case a smart device or modern camera with embedded GPS is used even the GPS coordinates the image was taken at are saved. All this information can be utilized in the rephotographic process and hence recapturing such a modern image is much more easy. Applications that utilize all this information are required, but most parameters can simply be readout. These include the approximate location of the original image as well as the intrinsic parameters of the original camera. Besides, if the photographer is interested in duplicating day time or season, these are directly accessible. Furthermore, knowing the intrinsic parameters of the original camera can significantly stabilize and speed up the process of extrinsic parameter estimation. Consequently, rephotography has the potential to become less challenging in the future.

On the other hand, in the future as well as at present new historic images are always recovered. These will never provide all this additional information. Furthermore, especially considering private images it is unlikely, that in 20 to 50 years all of these are available in electronic format. Even nowadays, many images are printed out to present to family and friends. Their electronic counterparts are likely to be deleted or get lost across greater time spans, while an old photo album is often kept as a memory. Consequently, the additional electronic information is lost as well and the need to deal with these images as with current historic imagery persists.

3.2 Related Work beyond Rephotography

In the following research beyond rephotography, but nonetheless related to this work is depicted. This includes several works concerned with images changing over time. Additionally research on multiple domain image matching as well as image retrieval is summarized and its relevance to this work is discussed.

3.2.1 Visualization of Time Change

Up to today a great amount of researchers is engaged in research on imagery, including similarity detection as well as 3D reconstruction from single images to image collections. Yet, most research focuses on modern imagery, while there are hardly any works concerned with older or historic images. In the following the few existing works related to the visualization of time change are presented.

4D Cities Project [Schindler and Dellaert, 2012]

First of all, there is the *4D Cities Project* of Schindler and Dellaert [2012]. Given a collection of historical images of one city they build a 4D model of that city, featuring time as the fourth dimension. To do so, SfM techniques are applied to construct a 3D model of the respective city, similar as in Agarwal et al. [2011] and Snavely et al. [2006]. Yet, Schindler and Dellaert [2012] add the component of time to their model by generating a time stamp for each image and a time interval for each shown object. Thus, their model allows a single view to change over time as objects begin and cease to exist.

In one of their previous works Schindler and Dellaert [2010] used a completely automatic approach to create such a 4D city model. In the initial step they use traditional SfM with SIFT features [Lowe, 2004] for 3D reconstruction. Only after this, the dimension of time is added to the model. However, they note that given a collection of 490 images of Atlanta taken from the 1930s to the 2000s their automatic approach was able to register only 102 images spanning 1956 to 1975. This is due to the great changes in appearance that took place over time. In his thesis Schindler [2010] notes that the number of feature correspondences between to images significantly decreases as the time span between capture increases, see Figure 3.14. As illustrated in Figure 3.15, this often leads to separate models of one city spanning different time periods.

Hence, Schindler and Dellaert [2012] expanded their software to use manual correspondences. With these they were able to construct a 4D model of Atlanta including 212 images from 1864 to 2008. Yet, they admit that even though manual construction is very accurate it takes its time. With their software a non-expert student unfamiliar with the city required 10 hours to create a 4D model of Seoul consisting of 88 images and depicting 29 buildings. Thus, they conclude that designing time invariant features is a major challenge for future work.

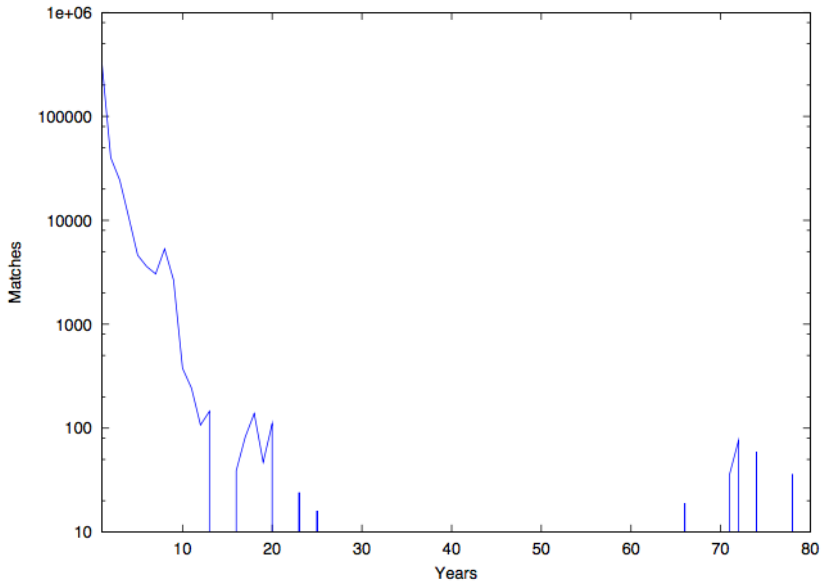


Figure 3.14: Plot of feature matches across time for images of Manhattan [Schindler, 2010]. Displayed is the number of geometrically consistent SIFT matches against the time span between image capture. Gaps result from a lack of matches as well as the scarcity of the image collection. Nevertheless, the plot clearly indicates the inverse relation ship between the number of correspondences and the time span between an image pair.

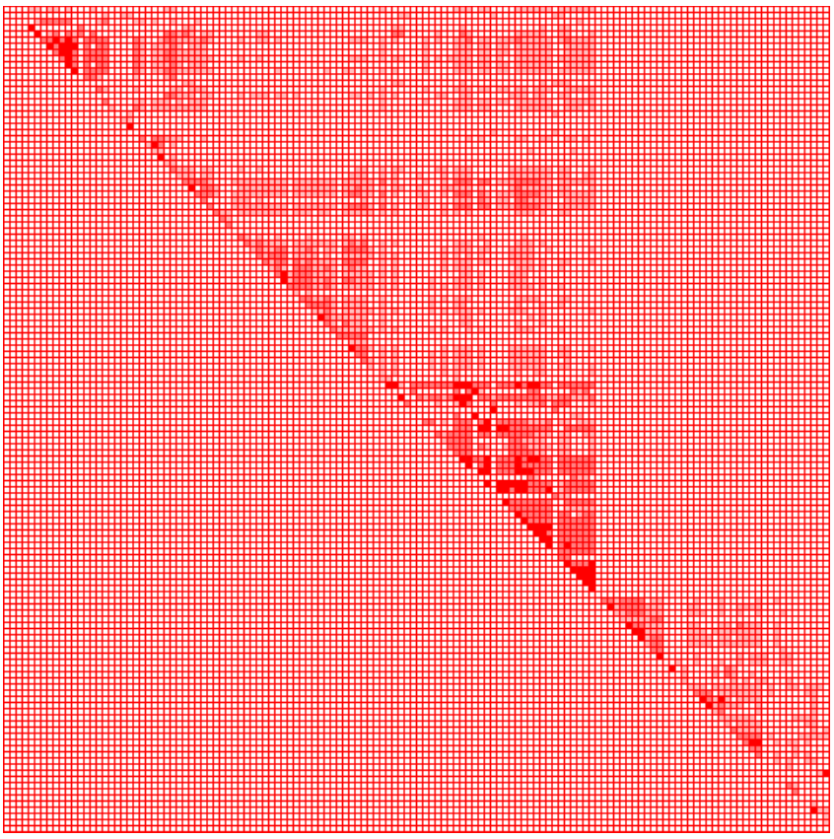


Figure 3.15: Match Table for Modern and Historical Images of Atlanta [Schindler, 2010].
The darker the squares the more geometrically consistent SIFT matches are found between the image pair, while white squares depict a number of matches below threshold. In the table two triangular structures are visible. The top one comprises matches between modern images while the bottom one is composed of historical image matches. Note that no matches link both triangles. Consequently using automatic correspondences only two separate 4D city models can be constructed.

Timescape Creation [Ali and Whitehead, 2016]

Ali and Whitehead [2016] present a method to create timescapes from modern and historical images of a historic building. In this context, their earlier works [Ali and Whitehead, 2014; Wolfe, 2013] evaluate the suitability of classic feature detectors and descriptors for matching historical and modern images. Their test dataset includes photos of popular sights such as the Eiffel Tower spanning several years.

In Wolfe [2013] Harris Corner Detector [Harris and Stephens, 1988], Good Features to Track [Shi and Tomasi, 1994], SIFT and Speeded Up Robust Features (SURF) [Bay et al., 2006] are compared. Even though altogether SIFT and SURF outperformed Harris Corner Detector and Good Features to Track, Wolfe [2013] comes to the conclusion that all tested detectors and descriptors perform poorly in matching historical to modern images. In detail, SIFT and SURF perform well if images are only few decades apart. However, with regard to long term image matching he draws the same conclusion as Schindler and Dellaert [2012]. As the time span between images increases classic image matching techniques frequently fail.

The tested feature detector and descriptor pairs in Ali and Whitehead [2014] include the following combinations: SIFT/SIFT, SURF/SURF, Harris/Harris, Binary Robust Indepen-

dent Elementary Features (BRIEF)/BRIEF [Calonder, 2010], Binary Robust Invariant Scalable Keypoints (BRISK)/BRISK [Leutenegger et al., 2011], Oriented FAST and Rotated BRIEF (ORB)/ORB [Rublee et al., 2011] and ORB/SURF. Using any of the above feature combinations, brute-force matching results include many outliers. To remove these Ali and Whitehead [2014] apply a Disparity Gradient Filter (DGF) based on the disparity gradient measure used in stereo vision [Trivedi and Lloyd, 1985]. The DGF discards false matches based on the deviation of the corresponding points motion vector from the median motion vector of all corresponding points, for more details please refer to Section 4.3.4. Altogether the combination of ORB detector and SURF descriptor achieved the best results and clearly outperformed the standard combinations, while SIFT/SIFT and SURF/SURF showed good performance. In contrast all other combinations performed poorly. However, one needs to take a closer look at the individual matching results of the whole dataset [Ali, 2016]. These reveal that even the features showing a high performance on average, are mostly successful in matching images taken only few years apart. True matching of historic and modern images separated by a large time span instead often fails, even with the ORB/SURF combination. Thus, the authors classify all evaluated feature descriptors and detectors as insufficient for the purpose of modern to historic image matching.

Finally, in Ali and Whitehead [2016], they propose to compute optical flow for densely sampled SIFT features to align images and create a timescape. This dense optical flow computation allows an inverse pixel by pixel warping to align images. Overall, this approach generates impressive results and timescape creation succeeds for their data. Unfortunately, they only present detailed results for succeeding images of each location. Yet, apart from few exemptions these succeeding images only span time periods of 1 to 10 years. For timescape creation it is sufficient to align only these images neighboring each other on the timeline. However, this way the need to develop an approach that succeeds in matching historical and modern images captured many years apart is avoided.

HistStadt4D

Since 2016 a research project at the Technical University Dresden and the University of Würzburg, focuses on visualizing Urban History in four dimensions. Its main goal is the georeferencing of historical images in 3D city models, to offer researchers a location based access to photos of digital image archives, which can so far only be searched by keywords [Niebling et al., 2018]. This shall give researchers better access to the changes a scene experienced and ease research in architectural history. A further goal is to create a mobile augmented reality application, which allows users to browse through historical photographs of a single location and provides additional historical knowledge on such [Maiwald et al., 2018a].

So far the research group concentrated on registering historic images of Dresden to a large 3D model of the entire city. In this context they faced similar challenges as Schindler and Dellaert [2012]. These include the sparsity of historic images available for alignment as well as the inability of classic feature matching approaches including SIFT [Lowe, 2004] to align historic to modern images [Niebling et al., 2018].

For automatic image alignment they presented an approach that matches historic images based on quadrilaterals [Maiwald et al., 2018b]. They apply the Canny Edge Detector [Canny, 1986] and detect closed contours via "border following". Afterwards, the Douglas-Peucker algorithm [Douglas and Peucker, 1973] is used to keep only quadrilaterals among all detected closed contours. The quadrilaterals are described based on the geometric properties of themselves and neighboring quadrilaterals. Applied to urban images this approach detects windows on building facades, which are mostly successfully matched to each other, see Figure 3.16.

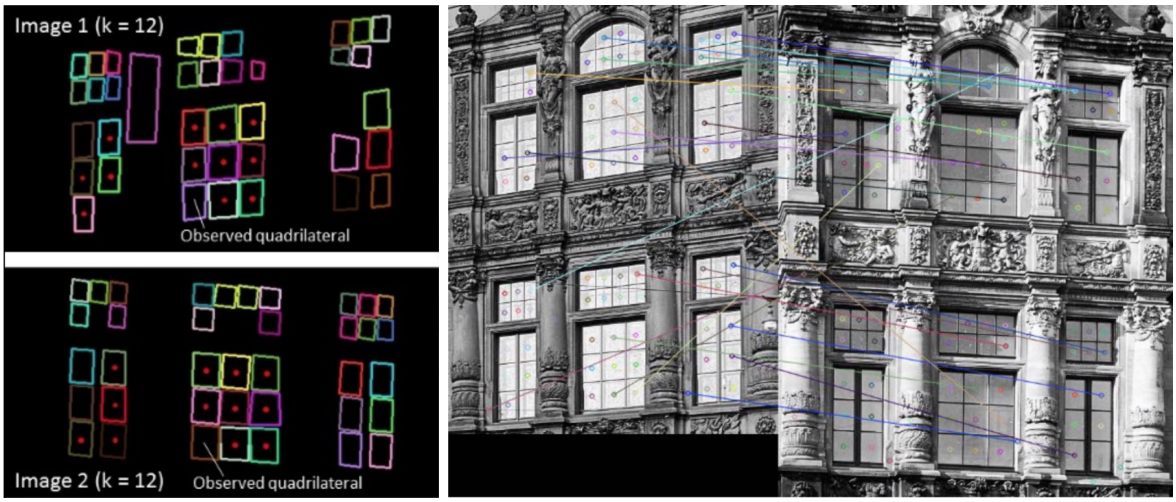


Figure 3.16: Illustration of feature matching based on quadrilaterals [Maiwald et al., 2018b].

However, the applicability of the presented approach is very limited. It can only be applied to historic images depicting facades with many quadrilaterals (windows) and even for these very few correct matches are established. For instance the comparison of the facades presented in Figure 3.16 only leads to 12 correct matches, despite the dense coverage by quadrilaterals. Furthermore, if buildings are captured from further away, many quadrilaterals are no longer identified, which results in even fewer correct matches. As a result, even for historic images containing facades with many quadrilaterals, regularly not enough matches for 6DoF pose estimation are identified [Maiwald et al., 2018b].

Otherwise the presented research mainly focuses on how to visualize historic photographs inside a 4D model and optimize the browsing experience for the user [Maiwald et al., 2019]. Besides possibilities to communicate research results to the public were discussed [Niebling et al., 2018]. However, these topics are less relevant in the context of this thesis.

Time Change in Dense Photo Collections

Matzen and Snavely [2014] use a large internet photo collection to reconstruct an urban scene and visualize its change in appearance across time. Compared to Schindler and Dellaert [2012], they focus on very huge densely sampled image collections that only span periods of 5 to 10 years. Consequently, they are able to use standard SfM methods utilizing classic features for automatic correspondence detection.

Instead of a large 4D model, Martin-Brualla et al. [2015] create a time-lapse video of a single scene to visualize time change. Their approach is fully automatic as well, but similar to that of Matzen and Snavely [2014], requires a very dense sampling of images. To achieve this they take advantage of the vast amount of images of popular sights available online. Hence, their videos comprise time spans of the current century from 4 to 10 years and are limited to certain regions.

Photobios [Kemelmacher-Shlizerman et al., 2011] allows the creation of time-lapse videos of faces visualizing their change in appearance across time. Even though results are impressive, the presented approach is specialized for and limited to faces and consequently less relevant to this work.

3.2.2 Multiple Domain Matching

Closely related to this thesis is the topic of multiple domain image matching. This describes the matching of challenging image pairs with disparate appearances including paintings, drawings, day and night pictures, historic and modern images. In the works of Shrivastava et al. [2011] and Aubry et al. [2014], described above, multiple domain matching is performed as well. However, both use global HOG descriptors whose weights are learned via linear classification. In contrast, the works presented in the following focus on local features. This allows the extraction of individual feature correspondences for image alignment instead of only measuring global similarity between images.

Local Symmetry Features [Hauagge and Snavely, 2012]

Hauagge and Snavely [2012] propose to match challenging image pairs by local bilateral and rotational symmetry features. To do so, they detect image similarities across symmetry axes at different scales. Afterwards, each image patch is scored by the presence of horizontal, vertical and rotational symmetry. This way they develop two symmetry detectors: one based on intensities (SYM-I) and the other based on gradients (SYM-G). Furthermore, based on these measures a feature descriptor (SYMD) is developed.

To evaluate their feature detectors and their descriptor they construct a new dataset comprised of challenging image pairs. This includes rephotographs, images with different rendering styles and great illumination differences. Results show that both detectors, especially SYM-G, outperform the classic feature detectors SIFT [Lowe, 2004] and Maximally Stable Extremal Regions (MSER) [Mata et al., 2004]. Additionally, if a dense sampling is applied, their SYMD descriptor manages to outperform classic descriptors as well. Yet, only a combination of SYMD with SIFT outperforms the simple SIFT descriptor on the keypoints detected with their symmetry detectors.

The main drawback of Hauagge and Snavely's [2012] method is that it is only invariant to scale changes. With the current implementation rotation invariance is only given for rotational symmetry but not bilateral symmetry. However, rotational invariance could be accomplished by sampling more symmetry axes. Instead, perspective changes are not considered at all. Unfortunately, this limits the approach to very few image pairs. Furthermore, Hauagge and Snavely [2012] note that their descriptor works best for image pairs displaying dramatic illumination changes. On historic image pairs on the other hand good results were mixed and for several pairs of their dataset none of the evaluated approaches generated good results. Thus, developing effective local feature detectors and descriptors for these types of image pairs is still an open challenge.

Joint Spectral Correspondences [Bansal and Daniilidis, 2013]

Another approach for disparate image matching is presented by Bansal and Daniilidis [2013]. First of all, they compute the SIFT descriptor of densely sampled keypoints in both images at two different scales. These are used to compute a joint image graph of the image pair. Afterwards, the eigen-spectrum of this graph is used to detect MSER keypoints, which are matched across images via the SIFT descriptor.

Bansal and Daniilidis [2013] use the dataset of Hauagge and Snavely [2012] to evaluate the performance of their JSPEC feature detector and descriptor. Results show an improved repeatability of their detector as well as an impressive average precision for their descriptor. However, more detailed analyses [Bansal, 2015] reveal that slight changes in scale, rotation and

perspective have a significantly negative impact on the performance. Consequently, similar to the approach of Hauagge and Snavely [2012], the applicability of JSPEC is limited to very few image pairs.

3.2.3 Image Retrieval

Another area were some works have focused on historic and modern image association is image retrieval. The major goal of image retrieval is to group images into different categories. To do so, machine learning techniques are used, that learn to differentiate diverse image categories based on a large labeled collection. Afterwards, if presented with a new image, the algorithm should be able to correctly estimate its category. Location recognition is often defined as an image retrieval problem as well. This means the location of an image is retrieved via finding the most similar image in a large collection.

The approach of Shrivastava et al. [2011] may also be used for image retrieval. Yet, they note that currently it is to computationally intensive to be practical for interactive image retrieval. Besides, with the location recognition approach of Aubry et al. [2014] another example of image retrieval has already been presented.

Location Recognition over Large Time Lags [Fernando et al., 2015]

Another work dealing with historic and modern photographs is presented by Fernando et al. [2015]. Their goal is to recognize the location of an old photograph given modern labeled images from the internet. In doing so, they evaluate the suitability of several detectors, descriptors and image representations for this task. The tested detectors include Difference of Gaussians (DoG) [Lowe, 2004], Hessian Affine [Mikolajczyk et al., 2005] and dense sampling. As descriptors RootSIFT [Arandjelović and Zisserman, 2012] and Local Intensity Order Pattern (LIOP) [Wang et al., 2011] are analyzed and images are represented by Bag-of-Words (BOW) or Fisher Vectors (FV). Additionally, the following detector-descriptor pairs are evaluated: Self Similarity [Shechtman and Irani, 2007], Symmetry Features [Hauagge and Snavely, 2012] and Edge Foci detector together with Binary Coherent Edge descriptor [Zitnick, 2010].

Using BOW for image representation the combination of dense sampling and RootSIFT descriptor achieves the best results. Fernando et al. [2015] note: "Due to the huge difference in visual appearance of old and new images the interest points detected by DoG and Hessian Affine loose their informative value." However, the best performance is achieved by the combination of Hessian Affine detector, RootSIFT descriptor and FV. Hence, somehow Hessian Affine complements FV better than dense sampling. In comparison, the approach of Shrivastava et al. [2011] based on a combination of HOG features and Exemplar SVM performed less well.

Finally, Fernando et al. [2015] emphasize that the difference in appearance of historic and modern images causes a domain shift at image descriptor level. To overcome this they test several domain adaption methods and are able to improve their results. Still as a possible future challenge they propose to seek novel image representations that are more suitable for coping with large time lags between images.

24/7 place recognition [Torii et al., 2015]

Torii et al. [2015] address the challenge of large-scale place recognition facing major illumination (day, night) and structural changes. They discover that matching images that depict large appearance changes becomes easier if these are captured from approximately the same view

point. Thus, they propose the use of a dense database generated from Google street-view images to perform location recognition. Similar to Fernando et al. [2015] they emphasize that detectors such as DoG are not reliable when facing major appearance changes. Hence, they use dense sampling combined with RootSIFT for feature generation and synthesize views to further expand the database of street-view images. Overall, their approach shows good results of location recognition for their 24/7 Tokio dataset comprised of 1,125 mobile phone query images from different times of day.

Unfortunately, google street-view images and the mobile phone images from the dataset are all modern and do not cover longer time periods. Consequently, it remains an open question whether the methods presented by Torii et al. [2015] are applicable for the alignment of modern and historical images as well.

Discussion

The major goal and conditions of image retrieval and location recognition differ from the objective of this thesis. However, especially the detailed feature detector and descriptor evaluation of Fernando et al. [2015] provides hints for this work. Yet, the impact of using global image descriptors and classifiers trained on large datasets leading to improved results needs to be kept in mind.

3.3 Summary

As mentioned in the discussion from Section 3.1.2, two major challenges need to be solved for simplifying the acquisition and presentation of rephotographs.

1. An automatic way to detect similarities between the historic image and the modern view of a scene is required. This need is directly stated by Schindler and Dellaert [2012] and relevant to most of the presented scientific approaches [Bae et al., 2010; Lee et al., 2011; Shrivastava et al., 2011]. Thus, image features stable across the long time periods faced in rephotography need to be established.
2. Advanced methods to ease the creation of rephotographs, such as the one presented by Bae et al. [2010], as well as approaches to post-processing need to be further developed and made available to the public.

In Chapters 4 to 6 of this work the first problem is tackled, while Chapter 7 focuses on practical applications developed in conjunction with this thesis.

Chapter 4

Performance of Classic Detectors and Descriptors

Image matching is a well studied topic. Feature based matching approaches are widely used for image alignment as well as 3D reconstruction via SfM. In theory they are suitable for post alignment of rephotographic image pairs as well as for providing guidance during image acquisition as shown by Bae et al. [2010]. However, diverse opinions exist, whether classic feature detectors and descriptors such as SIFT [Lowe, 2004] and SURF [Bay et al., 2006] are able to identify similarities between modern and historic images and are suitable for registering these, more details follow in Section 4.1.

Unfortunately, no detailed analysis of the suitability of classic detectors and descriptors for registering modern and historic images, meeting the requirements of this thesis, exists. Instead, some works perform no evaluation at all [Bae et al., 2010; Schindler and Dellaert, 2012], while others use datasets not replicating the conditions in rephotography [Fernando et al., 2015; Hauagge and Snavely, 2012]. In the following a new evaluation is presented, whose major goal is to answer the following questions:

1. How far are classic feature detectors and descriptors suitable for matching pairs of historic and modern images?
2. Can advanced filtering or additional constraints compensate for a general low feature matching performance?
3. Are alternative registering techniques required to match this kind of image pairs?

The study is conducted on a new dataset presented in Section 4.2. The composition and results of the evaluation are presented in Section 4.3. Finally, a discussion with regard to the previous questions is presented in Section 4.5. Parts of this evaluation were published in Becker and Vornberger [2019].

4.1 Literature

The diversity of opinion on the suitability of classic detectors and descriptors for long term image matching already showed in the previous chapter. Bae et al. [2010] and Schindler and Dellaert [2012] agree that SIFT fails to match historic and modern images. Furthermore, Schindler and Dellaert [2010] illustrate the inverse relationship between the number of correspondences and the time span between image capture, remember Figure 3.14. However, both do not mention the evaluation of any other detectors and descriptors.

Wolfe [2013] on the other hand tested more feature detectors and descriptors drawing the same conclusions as Schindler and Dellaert [2012]. Ali and Whitehead [2014] instead, performing tests on the same dataset, claim high performance for at least some classic detector and descriptor combinations. However, taking a closer look at their results, matching performance decreases as the time span between images increases. Unfortunately, their presented results are not detailed enough to analyze the cause of failure. Additionally, a shortcoming of their dataset is that it is comprised of views of popular sights, such as the Eiffel Tower. These age across years but experience hardly any structural changes.

Shrivastava et al. [2011] and Aubry et al. [2014] indicate that densely sampled HOG features with learned weights allow multiple domain matching, including the registration of historic to modern images. Fernando et al. [2015] state that the use of classic detectors as DoG and Hessian Affine is inadequate, but the combination of dense sampling and RootSIFT [Arandjelović and Zisserman, 2012] allows image retrieval. Yet, all these approaches generate and compare global image representations, instead of recovering full six degree of freedom camera poses, which requires local features. Thus, it remains unclear, whether global image representation and classifier training on large datasets are able to compensate for poor actual descriptor performance.

Torii et al. [2015] suggest that the difficulty of registering images showing major illumination and structural changes significantly decreases as perspective similarity increases. As Fernando et al. [2015] they stick to the combination of dense sampling and RootSIFT for location recognition. Yet, they do not show the applicability of their approach to image pairs spanning more than 15 years.

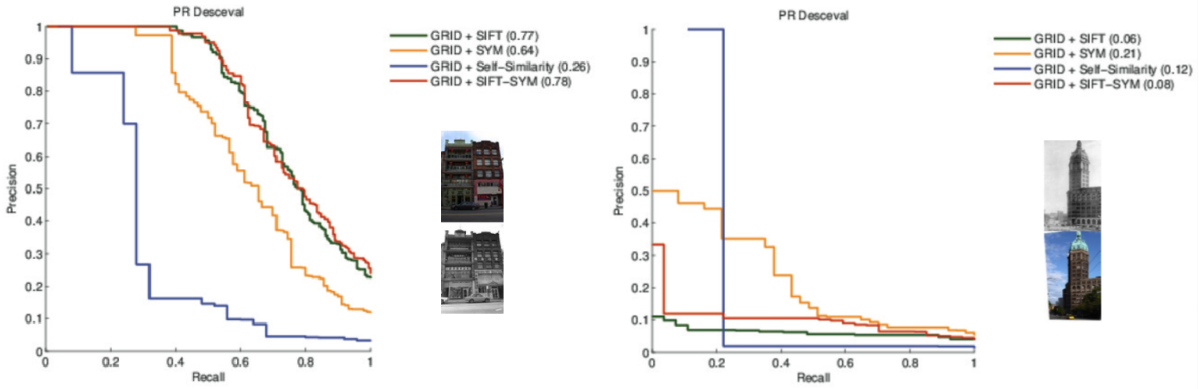
Hauagge and Snavely [2012] mention the shortcomings of classic feature detectors and descriptors in the context of multiple domain matching and propose the development of new symmetry based detectors and descriptors. However, their evaluation shows that a combination of the classic SIFT descriptor with dense sampling performs well on their dataset. This combination of dense sampling and SIFT is only outperformed by applying dense sampling and coupling the SYMD descriptor with SIFT descriptor, see Figure 4.1.

On the other hand, a closer look at the composition of their dataset[1] reveals that this consists of only very few historic and modern image pairs (<10, approximately 1/5 of the dataset). Instead, images of Notre-Dame (Paris) and the Painted Ladies (San Francisco) containing large illumination changes are dominant. Furthermore, the few historic-modern pairs included, depict a frontal view of a single building that aged across time, but shows no further structural changes. Of these the majority are matched well with the classic SIFT descriptor, while for two examples it fails completely, see also Figure 4.2. Consequently, because of the small number of historic and modern image pairs as well as their special structure, the question remains whether these results of Hauagge and Snavely [2012] generalize to other historic and modern image pairs.

	GRID	SIFT(DoG)	SYM-I	SYM-G
Self-Similarity	0.29	0.14	0.12	0.16
SIFT	0.49	0.21	0.28	0.25
SYMD	0.41	0.22	0.20	0.25
SIFT-SYMD	0.58	0.28	0.35	0.36

Figure 4.1: Mean average precision for different detector and descriptor pairs in Hauagge and Snavely [2012].

[1] `http://www.cs.cornell.edu/projects/symfeat/` (accessed on January 7th, 2020)



(a) Chinese Building **(b)** World Building

Figure 4.2: Precision and Recall Curves using dense sampling combined with different detectors for two historic and modern image pairs. The graphs show that for (a) the classic SIFT descriptor suffices, while in (b) it fails. [Hauagge and Snavely, 2012]

Gat et al. [2011]

In the literature further studies on the suitability of classic detectors and descriptors for long term image matching exist. Closely related to this work is the evaluation of Gat et al. [2011], who tested different detectors on historic rephotographs. They created a dataset including 73 rephotographs of various mountain landscapes that comprise differences in landscape, illumination and weather. The feature detectors evaluated include Harris Laplace [Mikolajczyk and Schmid, 2004], Hessian Laplace [Mikolajczyk et al., 2005], DoG [Lowe, 2004] and MSER [Mata et al., 2004]. For feature description SIFT is used and direct nearest neighbour search is performed.

While other studies commonly use *repeatability*, *precision* and *recall* [Bansal and Daniilidis, 2013; Hauagge and Snavely, 2012] to evaluate the performance of feature detectors and descriptors, Gat et al. [2011] apply a new measuring technique named *pass rate*. *Pass rate* requires minimal thresholds for *precision* and the *number of correct matches*. Then *pass rate* depicts the percentage of image pairs from the dataset that exceed the appointed thresholds. Hence, the measure incorporates the different amount of correct matches required for different applications. Furthermore, if comparing features for an entire dataset, pass rate is more meaningful than the average number of correct matches, since latter does not reflect the match distribution across the dataset. In case one half of the dataset features a great number of correct matches, while the other half contains almost none, the average number of correct matches is the same as if all pairs feature a medium number of correct correspondences.

In the final evaluation of Gat et al. [2011] the best performance is shown by the Hessian Laplace detector followed by DoG. For instance, requiring a minimum number of 10 correct matches Hessian Laplace achieved a pass rate of 60% at a precision of 10% discarding all matches with a distance threshold greater than 250. Harris Laplace and MSER on the other hand performed poorly.

Further Studies

Romaniuk et al. [2012] aim to create a 3D reconstruction of Rheims (France) based on old postcards spanning the 20th century. They analyze the suitability of different detector and descriptor combinations for matching postcards displaying temporal and structural modifications. Tested

detectors include Harris Laplace, Hessian Laplace, Harris Affine [Mikolajczyk and Schmid, 2004], Hessian Affine [Mikolajczyk and Schmid, 2004], SIFT and ASIFT, an affine invariant extension of SIFT [Morel and Yu, 2009]. Tested descriptors are Steerable filters [Mikolajczyk and Schmid, 2005], SIFT and PCA-SIFT [Ke and Sukthankar, 2004]. Overall, the combination of SIFT descriptor and detector shows the best performance with mean average precision ranging from 25% to 37%. Yet, their dataset only contains old postcards instead of historic and modern image pairs this work is interested in.

Valgren and Lilienthal [2010] study the impact of seasonal changes on SIFT and SURF upon localization in outdoor environments. They depict that U-SURF, the "upright" version of SURF performs most well on this task. Additionally, image localization rates can be further increased if an epipolar constraint is applied. However, their dataset only comprises images from a period of nine month. Hence, the applicability of their results to historic and modern image pairs which cover far greater time spans still has to be verified.

Stylianou et al. [2015] evaluate feature matching performance over time periods of five years. They use images from 20 outdoor webcams captured on a daily basis. These display different weather conditions, illumination and slight structural changes. The tested detectors include DoG and MSER, whereas tested descriptors are SIFT, SURF and DAISY [Tola et al., 2010]. Their analysis depicts that all detector and descriptor combinations show the same trends. Performance decreases significantly upon changing weather conditions. Thus, independent of their time span two sunny images match better than a pair comprised of a sunny and a cloudy image. The same effect is observed across seasonal changes. In general, even if taken five years apart two images from the beginning of the year show more correct correspondences, than one from January and one from August of the same year. The influence of structural changes on the other hand is tiny compared to changes in weather and lightning, at least for the considered time period of 5 years. Furthermore, Stylianou et al. [2015] analyze that the main cause of failure is not feature description but feature detection. This means for many keypoints no corresponding keypoint at the same location is detected in the reference image.

Discussion

With regard to the objections of this thesis the studies presented in the literature so far are not sufficient due to the following reasons. Many of the works mentioned suggest that matching older to modern images is more difficult but not impossible using classic feature detectors and descriptors [Ali and Whitehead, 2016; Aubry et al., 2014; Fernando et al., 2015; Hauagge and Snavely, 2012; Romaniuk et al., 2012; Shrivastava et al., 2011; Stylianou et al., 2015; Torii et al., 2015; Valgren and Lilienthal, 2010]. However, in all of them matching conditions were not equivalent to those faced by this thesis, due to the use of machine learning techniques on dense image collections, rather short time spans between image pairs, evaluation on images from multiple domains or a combination of these aspects. Furthermore, some of the authors pose the development of new more robust features for historic and modern image pairs as a challenge for future work [Fernando et al., 2015; Hauagge and Snavely, 2012]. Consequently, they suggest the need for more research in this area.

Other works [Ali and Whitehead, 2014; Schindler and Dellaert, 2012; Wolfe, 2013], who meet the conditions of this thesis more closely, claim that registering historical and modern images via classic features is infeasible. Yet, Schindler and Dellaert [2012] only access the suitability of SIFT features and apply a strict ratio threshold of 0.6 during correspondence search. As a result, their claim does not apply to classic feature detectors and descriptors in general. Wolfe [2013] and Ali and Whitehead [2014] on the other hand evaluate the performance of more detector

and descriptor pairs, but during the presentation of results image pairs of short and longer time spans are mixed. Thus, not many details of individual feature performance on historic and modern image pairs is provided.

Gat et al. [2011] performed a detailed evaluation of feature detectors on rephotographs. The shortcoming of their evaluation is that only the performance of diverse detectors but not different descriptors is assessed. Yet, other works have shown that specific detector and descriptor combinations may outperform others [Fernando et al., 2015; Hauagge and Snavely, 2012; Stylianou et al., 2015]. Furthermore, their dataset only includes rephotographs of mountain landscapes and none of urban scenes, which are popular in rephotography as well.

Finally, all of the presented studies apply default parameters for all detectors and descriptors. In addition, those who match historic to modern images [Gat et al., 2011; Hauagge and Snavely, 2012; Schindler and Dellaert, 2012] do not test any approaches to compensate for low feature matching performance. Only Ali and Whitehead [2014] apply an additional DGF to remove outliers. However, more advanced filtering or the use of additional constraints, similar to the epipolar constraint suggested by Valgren and Lilienthal [2010], may result in a performance boost of classic feature matching techniques.

Additional Questions for Evaluation

Due to these shortcomings of other studies, this work performs its own evaluation to analyze the suitability of classic detectors and descriptors for historic and modern image matching. Upon this the additional goal is to assess the following claims from the literature.

- Image registration becomes more difficult as the time span between images increases. [Schindler and Dellaert, 2012; Wolfe, 2013]
- Classic detectors loose their informative value for historic and modern image pairs, thus a dense sampling of keypoints is more applicable. [Fernando et al., 2015; Stylianou et al., 2015]
- Matching is easier for very slight shifts in viewpoint. [Torii et al., 2015]
- DGF is a useful tool for handling large amounts of outliers. [Ali and Whitehead, 2014]
- Additional constraints boost matching performance. [Valgren and Lilienthal, 2010]

4.2 Dataset

Since the datasets underlying the presented studies do not meet the conditions of this work [Hauagge and Snavely, 2012], are not publicly available [Gat et al., 2011] or both [Ali and Whitehead, 2014], we created a new dataset for the following evaluation. This contains 52 rephotographs of Manhattan from a collection created by Paul Sahner and published on his blog[2]. The collection comprises a variety of urban images including views of single buildings, whole streets as well as park areas, as shown in Figure 4.3. The modern photographs were taken between 2009 and 2013, while the originals span the entire 20th century. The distribution of time spans covered by the whole dataset is illustrated in Figure 4.4.

Taken by a professional rephotographer most of the image pairs of the collection are already well aligned. Yet, additionally we computed a homography between all image pairs based on the manual selection of corresponding points to further improve alignment. During this process we also rejected some image pairs of the original collection due to too few feature correspondences

[2] `http://www.nyc-grid.com` (accessed on April 30th, 2017)

Figure 4.3: Rephotographs from our dataset, including a view of a single building, a street corner lined with houses and a park. All image pairs feature high occlusion due to scene changes, show variations in lightning, and have been acquired by different cameras.

even among manual selection or the limitation of correspondences to single image regions. Some more image pairs were rejected as their view points could not be successfully aligned by the generated perspective transformation. The nyc-grid dataset is publicly available online[3] and was presented to the scientific community in Becker and Vornberger [2019].

We are aware, that the dataset only contains urban images and consequently does not represent the full range of rephotographic imagery. Ecologists for instance, who aim at the documentation of vegetation and climate change, usually take photographs in rural areas. A famous example is the rephotography project of the Northern Rocky Mountain Science Center usgs.gov, which depicts glacier melting in mountain landscapes. However, so far all related studies, apart from that of Gat et al. [2011], have focused on images of urban areas including man made structures.

This can be attributed to two main reasons. At first, archives contain more historic images of man made structures, including shoots of famous buildings or public and political events commonly taking place in more crowded urban areas. Second, with the exception of mountain contours, the appearance of most man made structures is more stable across large time spans than the vegetation of rural areas. Hence, it is assumed, that matching of landscape images containing hardly any man made structures is even more difficult. Consequently, as in related studies, the dataset only contains urban images, but the diversity of these urban images is ensured by including park areas and views of street corners lined with houses.

[3] `https://github.com/AnnKatrinBecker/nyc-grid-dataset` (accessed on January 7th, 2020)

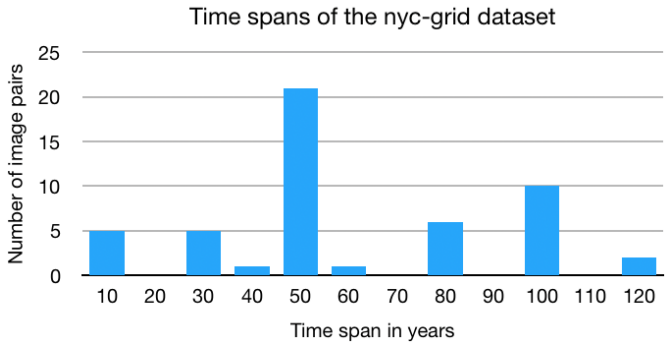


Figure 4.4: Illustration of the time spans covered by the nyc-grid dataset. Time spans are accumulated in steps of 10 years. Thus the first bar represents image pairs spanning 1-10 years, the second 11-20 years and so on.

4.3 Evaluation

In the following the results of the detailed evaluation based on the nyc-grid dataset are presented. This tests the suitability of classic detectors and descriptors for matching historic and modern image pairs as they arise during rephotography. A majority of previous studies revealed that such image pairs pose a challenge to classic detectors and descriptors. Hence, the evaluation aims not only at testing general performance but also at detecting measures to improve it. These measures include varying detector and descriptor parameters, applying different match filtering techniques for outlier reduction, the use of additional constraints, as well as a variation of approaches to detect outliers during final alignment computation.

It would be time consuming to compare all these measures for a large set of detector and descriptor combinations. Yet, we would like to analyze the suitability of a great variety of detector and descriptor combinations to ensure a broad validity of the study. Thus, we use a two step approach. At first, an initial evaluation is performed in which a large amount of detector and descriptor combinations is applied to the dataset using default parameters. Afterwards, matching performance is improved for the top detector and descriptor pairs. This way we ensure to include the best performing detector and descriptor pairs in the detailed evaluation.

During the detailed evaluation we test (A) the influence of the total number of detected keypoints as well as the influence of descriptor length. (B) We apply different match filtering techniques for outlier reduction including the ratio test, a maximal distance threshold and DGF [Ali and Whitehead, 2014]. (C) We use a variety of approaches for final alignment computation of all image pairs, which are able to handle different amounts of outliers. Finally, the robustness of the top performing detector and descriptor combinations against scale and view point changes is assessed as well as the applicability of our results to challenging image pairs beyond rephotography.

The tested set of detectors and descriptors includes those mentioned as successful in the literature. Among them dense sampling [Fernando et al., 2015; Hauagge and Snavely, 2012; Torii et al., 2015], as an alternative to a classic detector, RootSIFT [Fernando et al., 2015; Torii et al., 2015] and the combinations of SURF/U-SURF [Valgren and Lilienthal, 2010], SIFT/SIFT [Hauagge and Snavely, 2012; Romaniuk et al., 2012] and ORB/SURF [Ali and Whitehead, 2014]. An overview of all tested combinations is given in Section 4.3.2.

The next section discusses evaluation criteria. Afterwards, follow the results from the initial (Section 4.3.2) as well as the detailed evaluation (Sections 4.3.3 to 4.3.6).

4.3.1 Evaluation Criteria

Related studies use *repeatability*, *precision*, *recall* [Hauagge and Snavely, 2012] and *pass rate* [Gat et al., 2011] to evaluate feature detector and descriptor performance. We apply some of these, while others are not suitable in the context of this work. Furthermore, we develop a new criterion to rate the results of final image alignment.

Repeatability is used to assess feature detector performance. For each detected keypoint it measures whether a corresponding keypoint at the same location and of similar size exists in the other image, while keypoint appearance is not taken into account. In detail, if both images are related via a homography, the keypoints of the first image are transposed by this and the overlap with keypoints in the other image is computed [Mikolajczyk et al., 2005]. In practice repeatability represents the number of matches that should be returned by an ideal matching process. However, this assumes that the scene contains no occlusion or major appearance changes, since repeated keypoints in these areas do not represent the same scene content and effectively cannot and should not be returned as matches.

Historic and modern image pairs suffer from many structural changes. As structures disappear features to repeat disappear with them, while some may be replaced by accidental correspondences. Both of this has an undesired effect on the repeatability rate that can not be controlled without manual user annotation of each image pair. Consequently, for historic and modern image matching the repeatability measure, representing the number of matches that should be returned, loses its validity.

Recall is the number of correct matches divided by the number of matches that should have been returned. The matches that should have been returned comprise all those repeated during keypoint detection. Hence, recall is not a meaningful evaluation criterion for rephotographic image pairs for the same reasons as repeatability.

Precision is the number of correct matches divided by the total number of returned matches. Thus, it depicts the ratio of inliers and outliers among all corresponding keypoints returned by the matching process. If both images are related via a homography the number of correct matches can be determined via the overlap of corresponding keypoints after projection [Mikolajczyk and Schmid, 2005]. Precision is an important criteria in this evaluation, since successful image alignment requires certain rates of precision. More details on required precision rates for matching are found in Section 4.3.5.

Average Precision depicts the average precision of all images of the entire dataset. In many studies this value is used to compare detector and descriptor performance. Unfortunately, average precision does not provide any information on the precision distribution. It can be boosted by raising the precision of a few images of the dataset to very high values, while neglecting low precision values of another part of the dataset. This is not desirable, since we aim at registering the entire dataset, and the major goal is raising precision for difficult images, featuring low initial precision values.

Pass rate [Gat et al., 2011] is calculated on the entire dataset and based on precision and the number of correct matches. As mentioned in Section 4.1 it requires minimal thresholds for precision and the number of correct matches. Then pass rate is the percentage of image pairs from the dataset that exceed the appointed thresholds. Consequently, the measure incorporates the different amount of correct matches required for different applications. Especially if in a dataset few images show a high number of correct correspondences, while others hardly contain any, pass rate is more meaningful than the average number of correct matches. As a result, together with average precision we mostly use pass rate to compare results in this evaluation.

Evaluating Generated Homographies. To the author's knowledge there is no well-established

standard to evaluate the quality of an automatically estimated homography. We developed a new measure accessing estimated homography quality based on comparing the projections generated by the estimated and ground truth homography. In detail, we project the image center and all image corners and calculate the mean distance between both projections for each point pair. Additionally, the surface area of both projections is compared. Given these values we establish a strong and weak thresholds to decide whether an estimated homography results in good, adequate or bad alignment. For more details please refer to Section 4.3.5.

We do not consider computation time as a criterion in this evaluation, since especially in the context of post-processing rephotographs we do not face any time constraints . Yet, our evaluation set contains fast binary as well as more computational expensive floating point descriptors. Thus, depending on the requirements of his application the reader may choose the best feature detector and descriptor pair among the compared once.

4.3.2 Initial Evaluation of Diverse Detector and Descriptor Combinations

The initial evaluation assess the performance of diverse feature detectors and descriptors on matching historical to modern image pairs. Previous studies report that it is critical to find enough correct matches at all for these image pairs [Schindler and Dellaert, 2012]. Hence, we first perform a direct nearest neighbour search without applying any filters such as the ratio test or a maximal distance threshold, since these eliminate correct matches as well. We are aware, that applying no filtering mechanism eliminating false correspondences leads to poor precision values. Therefore, in the initial evaluation we only use the average number of correct matches to compare feature detectors and descriptors with each other, while we expect to improve precision in the following detailed evaluation.

The detectors compared in this section include DENSE10 (dense sampling with a step size and a keypoint radius of 10), DENSE25 (step size and radius equal 25 as proposed in Hauagge and Snavely [2012]), MSER [Mata et al., 2004], Features from Accelerated Segment Test (FAST) [Rosten and Drummond, 2006] and CenSurE [Agrawal et al., 2008]. Evaluated descriptors are U-SURF [Bay et al., 2006], RootSIFT [Arandjelović and Zisserman, 2012], BRIEF [Calonder, 2010], Fast Retina Keypoint (FREAK) [Alahi et al., 2012], DAISY [Tola et al., 2010] and Learned Arrangements of Three Patch Codes (LATCH) [Levi and Hassner, 2016]. While the following features are used as detectors and descriptors: SIFT [Lowe, 2004], SURF [Bay et al., 2006], ORB [Rublee et al., 2011], BRISK [Leutenegger et al., 2011], KAZE [Alcantarilla et al., 2012], and AKAZE [Alcantarilla and Solutions, 2011]. We aimed at including all detectors and descriptors in the test set that have been reported to show good performance in related studies, including dense sampling [Fernando et al., 2015; Hauagge and Snavely, 2012; Torii et al., 2015], RootSIFT [Fernando et al., 2015; Torii et al., 2015] and the combinations of SURF/U-SURF [Valgren and Lilienthal, 2010], SIFT/SIFT [Hauagge and Snavely, 2012; Romaniuk et al., 2012] and ORB/SURF [Ali and Whitehead, 2014]. Afterwards, we extended the set with prominent detectors such as MSER and FAST as well as features only recently proposed and not widespread yet such as LATCH and AKAZE. Furthermore, the test set includes fast binary as well as more computational expensive floating point descriptors. For more details on a certain feature detector or descriptor please refer to the original paper.

We use the OpenCV[4] implementation of all these features including their respective default parameters. Only for ORB we limit the maximum number of keypoints to 10000 instead of 500 for the initial evaluation. In the initial evaluation we apply all possible combinations of these

[4] `http://www.opencv.org` (accessed on January 7th, 2020)

Detector	Descriptor	TotalMatches	CorrectMatches	Precision (%)
FAST	**Avg**	13432.5	188.0	1.28
ORB	**Avg**	8051.2	167.1	2.02
SURF	**Avg**	5221.8	92.2	1.64
DENSE10	**Avg**	5718.8	79.8	1.41
BRISK	**Avg**	7311.9	78.7	1.20
AKAZE	**Avg**	2893.4	34.2	1.29
DENSE25	**Avg**	841.8	31.8	3.94
KAZE	**Avg**	2692.6	21.8	0.87
SIFT	**Avg**	4022.5	9.5	0.24
MSER	**Avg**	745.2	9.4	1.35
STAR	**Avg**	829.7	9.0	1.15

Table 4.1: Overview of average detector performance.

Detector	Descriptor	TotalMatches	CorrectMatches	Precision (%)
Avg	**SIFT**	4933.2	161.5	3.9
Avg	**LATCH**	4660.3	88.2	2.0
Avg	**BRIEF**	4645.0	83.9	1.8
Avg	**ORB**	4674.5	69.5	1.6
Avg	**U-SURF**	4933.2	66.7	1.5
Avg	**BRISK**	4651.3	47.9	1.0
Avg	**DAISY**	5344.0	39.8	0.8
AKAZE	**AKAZE**	2894.5	38.7	1.38
Avg	**SURF**	4933.2	37.6	0.8
KAZE	**KAZE**	2826.0	27.9	1.01
Avg	**FREAK**	4040.9	2.4	0.1

Table 4.2: Overview of average descriptor performance.

feature detectors and descriptors to the dataset. Note that some detector and descriptors can not be combined. For instance, AKAZE detector can only be used with AKAZE descriptor. For all feature combinations nearest neighbour search is performed using the hamming distance or L2 norm, depending on the descriptor used.

Comparison of Feature Detectors and Descriptors

At first we assess detector performance. Comparing detector repeatability values is not an option as explained previously (Section 4.3.1). Instead, we compute the average number of correct matches of all descriptors combined with a single detector. The results are depicted in Table 4.1. Considering the average number of correct matches FAST and ORB clearly outperform all other detectors. Further behind follow SURF, DENSE10 and BRISK. All following descriptors feature an average number of correct matches below 35. Yet, we need to keep in mind that poorly performing descriptors in the test set have a negative impact on these average values.

If a similar approach is used to evaluate descriptor performance, we receive the results of Table 4.2. Here the SIFT descriptor shows the highest performance followed by LATCH, BRIEF, ORB and U-SURF. Again detector performance highly influences average descriptor performance. However, since a similar set of detectors is used for all descriptors results are somehow comparable.

Detector	Descriptor	Total Matches	Correct Matches	Precision (%)	Pass Rate 8 (%)	Pass Rate 16 (%)
FAST	**SIFT**	14094.1	491.0	3.32	98	92
FAST	**LATCH**	12768.2	365.4	2.54	88	81
ORB	**SIFT**	8667.2	333.0	3.87	96	96
ORB	**U-SURF**	8667.2	268.5	3.13	94	88
DENSE10	**SIFT**	6033.7	262.9	4.58	96	88
FAST	**BRIEF**	12713.9	248.7	1.74	87	83
FAST	**ORB**	12553.9	247.2	1.77	87	79
SURF	**SIFT**	5538.7	221.7	3.90	92	90
ORB	**BRIEF**	8667.2	199.8	2.36	88	83
ORB	**BRISK**	7441.5	185.4	2.54	90	79
BRISK	**SIFT**	7479.4	164.0	2.32	81	75
ORB	**SURF**	8667.2	155.8	1.79	90	71
DENSE10	**LATCH**	5517.2	145.8	2.70	87	83
ORB	**LATCH**	8667.2	128.5	1.48	83	73
BRISK	**BRIEF**	7309.0	127.7	2.01	81	79
SURF	**BRIEF**	5291.1	123.2	2.23	88	79
ORB	**ORB**	8667.2	122.3	1.41	87	75
SURF	**U-SURF**	5538.7	118.8	2.09	88	77
DENSE25	**SIFT**	903.8	112.7	13.37	94	83
FAST	**U-SURF**	14094.1	103.3	0.67	81	73
BRISK	**U-SURF**	7479.4	100.1	1.62	85	79

Table 4.3: Overview of top 22 descriptor and detector combinations sorted by the number of correct matches.

Considering these results the combination of FAST detector and SIFT descriptor is most successful in matching historic to modern images. Furthermore, combinations such as ORB/SIFT, SURF/SIFT, ORB/LATCH and ORB/BRIEF are expected to show good performance. Yet, other works have shown that specific detector and descriptor combinations may outperform others [Fernando et al., 2015; Hauagge and Snavely, 2012; Stylianou et al., 2015]. Thus, we rank all applied feature detector and descriptor combinations by their average number of correct matches calculated on all image pairs of the nyc-grid dataset. An overview of the top 22 detector and descriptor pairs with an average of more than 100 correct matches is given in Table 4.3.

This comparison shows that indeed FAST/SIFT is the best feature combination. Besides, as expected FAST, ORB, DENSE10 and SURF are descriptors appearing in combinations showing good performance, while SIFT, LATCH, U-SURF and BRIEF are represented detectors. Yet, also DENSE25 in combination with SIFT produces good results, despite a low average performance among all detectors. Additionally, as anticipated single feature combinations show higher average numbers of correct matches.

However, as mentioned previously the average number of correct matches does not provide any information on their distribution among different image pairs. Since we expect to find only very few correct matches for certain images this distribution is extremely relevant for this evaluation. As a result, we compute pass rates for these top detector and descriptor combinations requiring a minimal number of 8 and 16 correct correspondences, while imposing no threshold on precision. We choose 8 and 16 as minimal values, since 8 points are generally required to estimate the fundamental matrix for 3D reconstruction, while 16 is the required minimal number of correspondences suggested by Schindler and Dellaert [2012].

Detector	Descriptor	Total Matches	Correct Matches	Precision (%)	Pass Rate 8 (%)	Pass Rate 16 (%)
ORB	**SIFT**	8667.2	333	3.87	96	96
FAST	**SIFT**	14094.1	491	3.32	98	92
SURF	**SIFT**	5538.7	221.7	3.9	92	90
ORB	**U-SURF**	8667.2	268.5	3.13	94	88
DENSE10	**SIFT**	6033.7	262.9	4.58	96	88
FAST	**BRIEF**	12713.9	248.7	1.74	87	83
ORB	**BRIEF**	8667.2	199.8	2.36	88	83
DENSE10	**LATCH**	5517.2	145.8	2.7	87	83
DENSE25	**SIFT**	903.8	112.7	13.37	94	83
FAST	**LATCH**	12768.2	365.4	2.54	88	81

Table 4.4: Overview of top 10 descriptor and detector combinations sorted by Pass Rate 16.

We use pass rate to reduce the set of top performing detector and descriptor pairs, by eliminating all combinations featuring a pass rate below 80 for >=16 correct matches. This leaves us with the top 10 combinations presented in Table 4.4. The further evaluation focuses on this set of top 10 feature detector and descriptor combinations.

Please note, considering the coverage of correct matches among the whole dataset (pass rate), the ranks of the different detector and descriptor combinations from Table 4.3 changed. The combination of ORB/SIFT shows the highest pass rates, while FAST/LATCH performs rather poor in terms of pass rate. In detail we have to expect that at least 20% of the image pairs of the dataset can not be matched successfully via FAST detector and LATCH descriptor. In the rest of the evaluation pass rate is the more important criteria.

SIFT vs. RootSIFT and FLANN vs. Direct Nearest Neighbour Search

The SIFT descriptor is very present among the top 10 detector and descriptor combinations, but its computation and direct matching is computationally expensive. Thus we evaluate the effects of applying Fast Approximate Nearest Neighbour Search (FLANN) [Muja and Lowe, 2009] instead of direct nearest neighbour search to match SIFT keypoints. Furthermore, RootSIFT [Arandjelović and Zisserman, 2012] a variant of SIFT, which is supposed to be more robust is tested.

FLANN constructs a fast search structure from the keypoints of one image. In our case this consists of four k-d trees, which can be searched in parallel. Afterwards, this search structure is used to find approximate nearest neighbors for the keypoints of the second image. FLANN does not necessarily find the nearest neighbour of a keypoint every time. Its precision can be improved by increasing the number of k-d trees as well as the number of times these should be traversed, yet this also takes more computation time.

The results of applying FLANN compared to direct nearest neighbour search to match SIFT keypoints are shown in Figure 4.5. Here only the performance for keypoints detected with ORB and FAST is displayed, but the other detectors among the top 10 including SURF, BRISK, DENSE25 and DENSE10 showed similar results. Both combinations show minor negative effects on pass rate and precision when FLANN instead of direct nearest neighbour search is applied. Yet, the reduction of the required computational cost of matching is immense. Furthermore, applying RootSIFT instead of SIFT compensates for some of this negative effects and is even faster. Hence, in the following evaluation RootSIFT instead of SIFT descriptor is used and it is matched with FLANN instead of direct nearest neighbour search to speed up the evaluation.

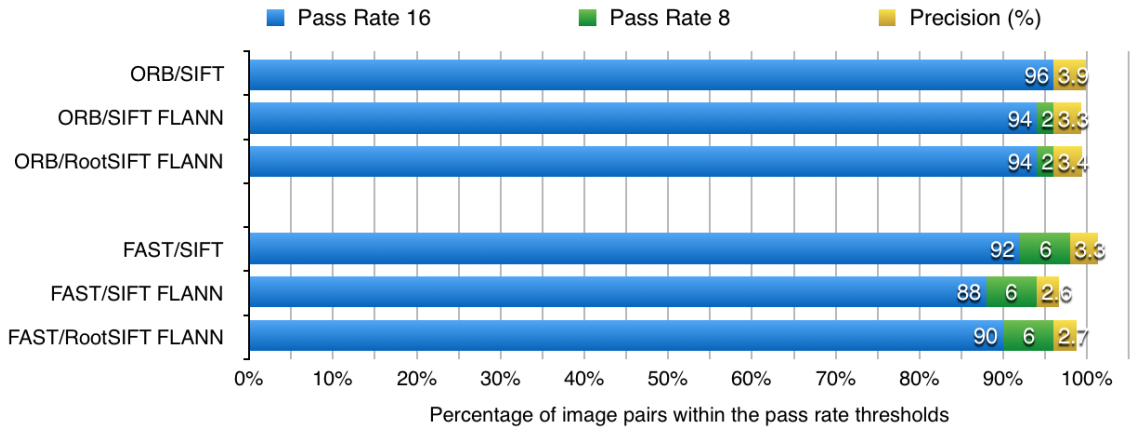


Figure 4.5: Diagram displaying the effects of applying FLANN instead of direct nearest neighbour search to match SIFT keypoints and comparison of SIFT with its variant RootSIFT.

4.3.3 A: Varying Detector and Descriptor Parameters

During the initial evaluation we found the top 10 feature detector and descriptor combinations. Yet, even for these top detector and descriptor combinations approximately 20% of the images of the dataset feature a very low number of correct matches. This we expect to drop even further as soon as we apply match filtering techniques for outlier reduction in Section 4.3.4. Thus, at first we try to further increase the number of correct matches via varying detector and descriptor parameters. Additionally, by varying descriptor parameters we also aim at increasing the low precision values faced in the previous section. Raising these is required to allow methods such as RANSAC [Fischler and Bolles, 1981] to estimate the correct alignment of an image pair, since at a reasonable computational expense these are only able to handle a certain amount of outliers. For more details please refer to Section 4.3.5.

Varying the Parameters of Dense Sampling

Among the detectors present in the top 10 combinations are DENSE10 (dense sampling with spacing and keypoint radius of 10 pixels) as well as DENSE25 (spacing and radius of 25 pixels). These initial variants of dense sampling were more or less chosen arbitrarily. Therefore, in the following we assess how varying keypoint spacing and scale influences performance. At first, we keep a spacing of 10 pixels and sample dense keypoints with radii of 25, 35 and 50 pixels. These are described and matched via RootSIFT and LATCH. Afterwards, we decrease the spacing to only 5 pixels and set the radius to 25, which showed the best results. This way we can analyze the effects of increasing the number of sampled keypoints.

Figure 4.6 shows the influence of changing the parameters of dense sampling on pass rate and precision for the entire dataset. Increasing the keypoint radius to 25 has a great positive effect on images featuring only very few correct correspondences. In detail, the number of images with at least 16 correspondences increased by approximately 10% for RootSIFT as well as LATCH descriptor. Instead, increasing the radius to 35 or 50 does not further improve the performance in terms of pass rate. Reducing spacing and therefore increasing the number of keypoints sampled has a great positive impact. The pass rate of dense sampling combined with LATCH descriptor can be increased to the maximum of 100% if sampling keypoints at a spacing of 5 instead of 10 pixels. On the other hand, sampling more keypoints does not have a significant negative effect on precision.

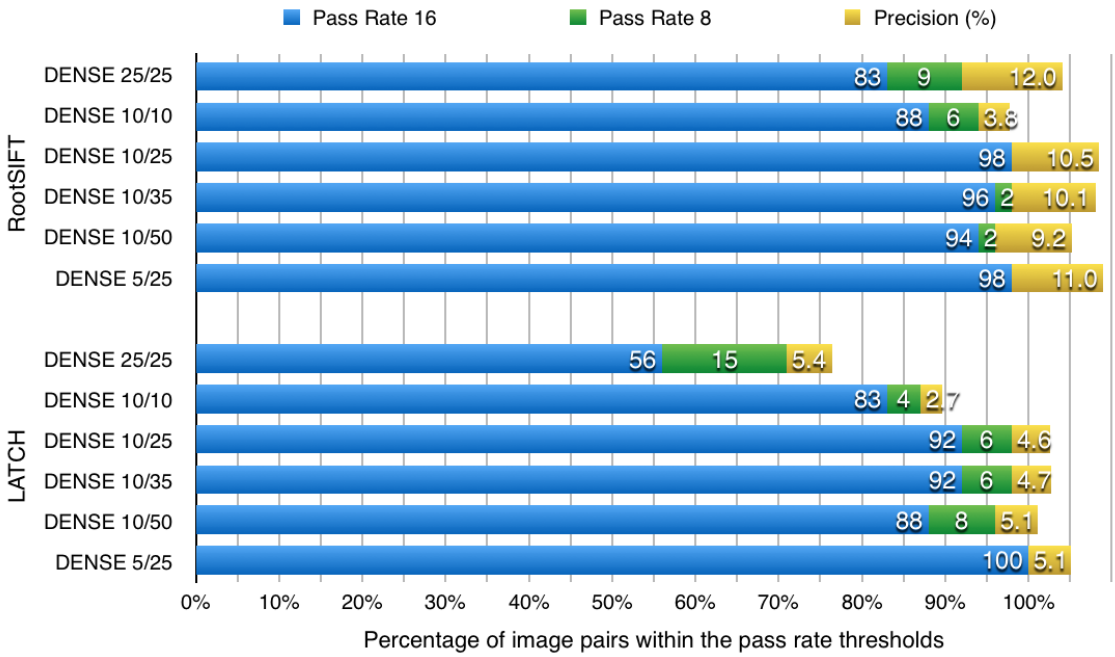


Figure 4.6: Diagram showing the effects of changing the parameters of dense sampling. For each dense sampling version combined with RootSIFT and LATCH descriptor the pass rates for at least 16 and 8 correct matches, as well as precision are depicted. The first number after each feature combination depicts the keypoint spacing, while the second number is the keypoint radius, both in pixels.

As a result, we decided to consider DENSE5:25/RootSIFT and DENSE5:25/LATCH in the further evaluation and eliminate DENSE10/SIFT, DENSE10/LATCH and DENSE25/SIFT from the test set. Yet, at the same time we keep in mind that dense sampling requires image pairs of similar scale and resolution. This is the case in our dataset, but not necessarily in practice. Furthermore, ideal keypoint size is likely related to image resolution. Hence, if images are much larger or smaller than in our dataset (mostly around 800x600 pixel) other radii may show better performances. However, in rephotography the resolution of images may be adapted via preprocessing and we can expect similar scales especially during post-processing. Thus, the application of dense sampling is an option.

Increasing the Number of Keypoints for ORB and SURF Detector

In the case of dense sampling we demonstrated that increasing the number of densely sampled keypoints boosts performance in terms of pass rate. For other detectors among the top 10 we directly or indirectly limited the number of keypoints during the detection process as well. For ORB we set the maximal number of returned keypoints to 10000 in the initial evaluation and for SURF the Hessian threshold was set to 100 by default. Thus, we check in how far increasing the number of keypoints for these detectors leads to an increase in pass rate. Since FAST already returned a very high number of keypoints, on average approximately 15000, we do not force a further increase for this detector.

We increase the number of returned keypoints for ORB up to 20000 and decrease the Hessian threshold for SURF down to 25. Figure 4.7 illustrates how this influences the number of correct matches found for the entire dataset. In general extending the number of detected keypoints increases the number of images containing at least 16 or 8 matches. Only for the combinations

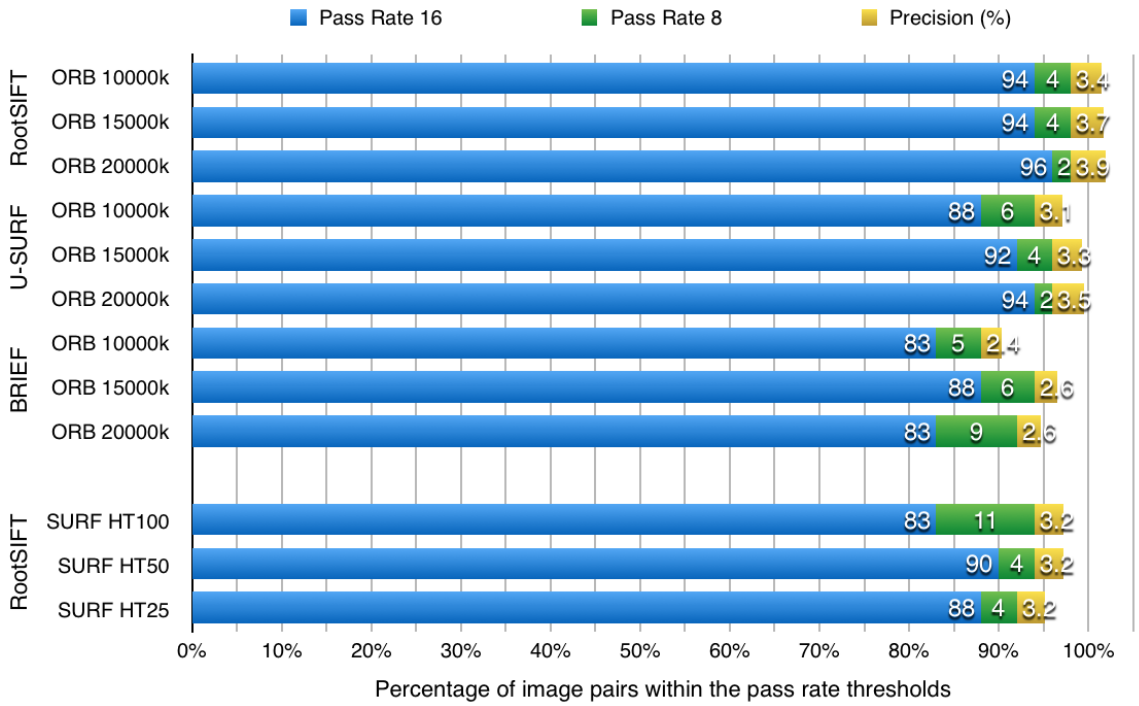


Figure 4.7: Illustration of the effects of increasing the number of keypoints for combinations including ORB and SURF detector.

ORB/BRIEF and SURF/RootSIFT too many keypoints decrease performance again. This is due to the fact that many keypoints also increase the probability to find false matches. This probability is even higher for a binary descriptor such as BRIEF, which does not have as much discriminative power as its floating point counter parts RootSIFT or U-SURF. Similarly, applying SURF with a very low Hessian threshold seems to generate to many similar keypoints leading to a decrease in matching performance.

Additionally, we need to consider precision. In fact, we expected precision values to significantly drop as the number of keypoints increases, due to a higher relative increase in the number of false correspondences. Instead, increasing the number of detected keypoints has only a minor impact on average precision and in most cases this is positive. Thus, in reality the relative increase between the number of keypoints and the number of correct matches is very similar.

Consequently, we use ORB20k/RootSIFT, ORB20k/U-SURF, ORB15k/BRIEF and set the Hessian threshold for SURF detector to 50 in the further evaluation.

Varying Descriptor Length

The previous results show that reducing the number of detected keypoints via filtering by response does not improve precision and additionally has a critical effect on the number of images showing very few correct correspondences. Thus, applying tighter response thresholds during keypoint detection does not make sense for matching historic to modern image pairs. Another approach to increase the number of correct matches and improve precision is to enhance the discriminative power of the applied descriptors. To do so, the number of elements a descriptor uses to describe a keypoint can be increased. This is accompanied by an increase in computational cost. In the following we evaluate the effects of raising the number of elements of U-SURF,

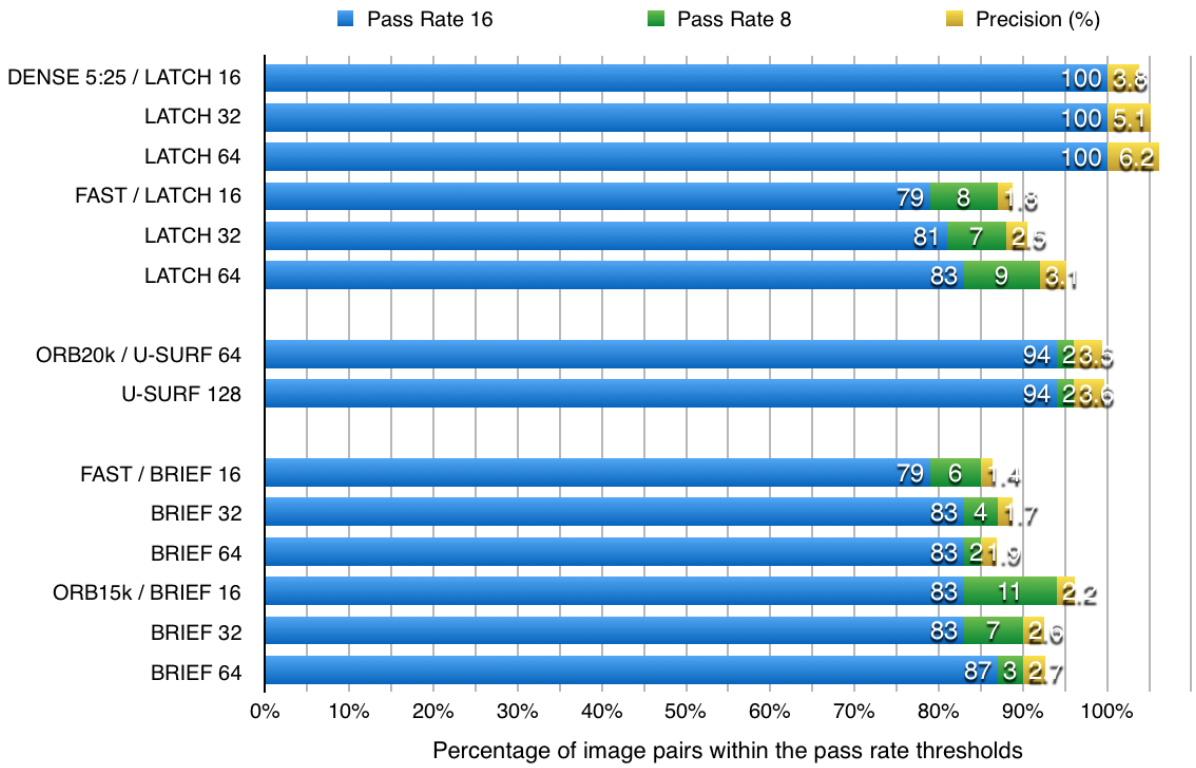


Figure 4.8: Illustration of the impact of descriptor length on matching performance.

BRIEF and LATCH descriptor, while we abstain from increasing the number of elements used for RootSIFT (128) since computing such is already time consuming. In detail the number of elements of the U-SURF descriptor is increased to 128 from 64, whereas for BRIEF and LATCH we use a 64 instead of 32 element descriptor. To verify the observed trends we additionally evaluate the results of BRIEF and LATCH descriptor with only 16 elements.

The outcome is displayed in Figure 4.8. For LATCH descriptor a clear trend is visible. Here an increased descriptor length is accompanied by an increase in pass rate as well as an increase in average precision values of up to 1%. Instead, changing the number of elements of the U-SURF descriptor has hardly any impact. For BRIEF descriptor results are more inconsistent. While for the combination ORB15k/BRIEF increasing the descriptor elements from 32 to 64 raises the number of images containing at least 16 correct correspondences, an increase for the combination FAST/BRIEF results in a decrease of the number of images containing at least 8 correct correspondences. The impact on average precision on the other hand is below 0.2% and thus not significant. Besides, the number of images containing at least 16 correct matches which is more relevant for our application remains stable. Thus, we decide to use LATCH and BRIEF with 64 elements instead of 32 in the further evaluation, while we stick with 64 elements for U-SURF.

Finally, we remain with the following top 9 detector and descriptor combinations for the further evaluation:

- DENSE5:25/RootSIFT
- DENSE5:25/LATCH64
- ORB20k/RootSIFT
- ORB20k/U-SURF
- FAST/RootSIFT
- SURF/RootSIFT
- FAST/BRIEF64
- ORB15k/BRIEF64
- FAST/LATCH64.

4.3.4 B: Outlier Reduction via Different Match Filtering Approaches

In the previous steps of the evaluation we have varied detector as well as descriptor parameters to increase the minimal number of correct matches. Now the goal is to increase precision values, which are extremely low for a great amount of image pairs in the dataset. A standard method to improve precision is to apply filters to all returned correspondences that eliminate false matches. In detail, we evaluate two common match filtering approaches namely the ratio test [Lowe, 2004] and maximal descriptor distance threshold. Furthermore, we test the performance of DGF [Ali and Whitehead, 2014], as well as the outlier filtering abilities of K-connected VLD-based matching (K-VLD) [Liu and Marlet, 2012]. The latter use constraints beyond descriptor distance to eliminate false matches.

Ratio Test

The ratio test was initially proposed by Lowe [2004]. It requires to determine two candidate matches for each keypoint, specifically the best two matches based on descriptor distance. Afterwards, the ratio between the distances of the first and second best match is calculated. The method assumes the following: If the distance to the second best match is significantly higher than that to the best match, the best match is a correct one, because it is by far the best choice. On the other hand, if both matches show similar distances the returned match is regarded to be false. Thus, matches featuring a high ratio are discarded.

Lowe [2004] recommend to eliminate matches featuring a ratio of more than 0.8 while others propose even stricter values of 0.6 [Schindler and Dellaert, 2012]. In this evaluation we apply ratios of 0.9, 0.8 and 0.6. The results for each of the top 9 detector and descriptor combination are shown in Figure 4.9. For each combination the influence on average precision, the percentage of images containing at least 16 correct matches, as well as different pass rates are shown. These pass rates always require a minimal number of 16 correct matches and additionally expect precision values of at least 1.0, 5.0 and 10.0. Consequently, all pass rate values (yellow, orange and red lines) are capped by the percentage of image pairs containing at least 16 matches (green line).

All graphs nicely illustrate that precision values increase as stricter ratios are applied. This shows, that in general the ratio test eliminates more false than correct correspondences. Yet, also many correct correspondences are discarded as the severe drop in the percentage of images containing at least 16 matches shows. Due to the different nature (illumination, capturing methods) of modern and historic images even correct matches often feature high distances. As a consequence the difference in distances between the best and second best match is rather low. As soon as a ratio of 0.8 is applied the percentage of images containing at least 16 correct matches drops below 60% for all detector and descriptor combinations. As a result, more than 40% of image pairs can not be matched due to a lack of correct correspondences, which is clearly not acceptable in practice.

Instead, applying a ratio threshold of 0.9 does not drastically decrease the number of images containing at least 16 correct matches. At the same time the percentage of image pairs featuring precision values above 5.0 significantly increases for most detector and descriptor combinations. Yet, for successful alignment usually precision values of at least 10% are required, see also Section 4.3.5. Consequently, other filtering approaches are required.

Figure 4.9: Results of applying the ratio test to the matches returned for the top 9 detector and descriptor combinations. For each the influence on average precision (blue line), the percentage of images containing at least 16 correct matches, as well as different pass rates are shown. All pass rate values (yellow, orange and red lines) are capped by the percentage of image pairs containing at least 16 matches (green line).

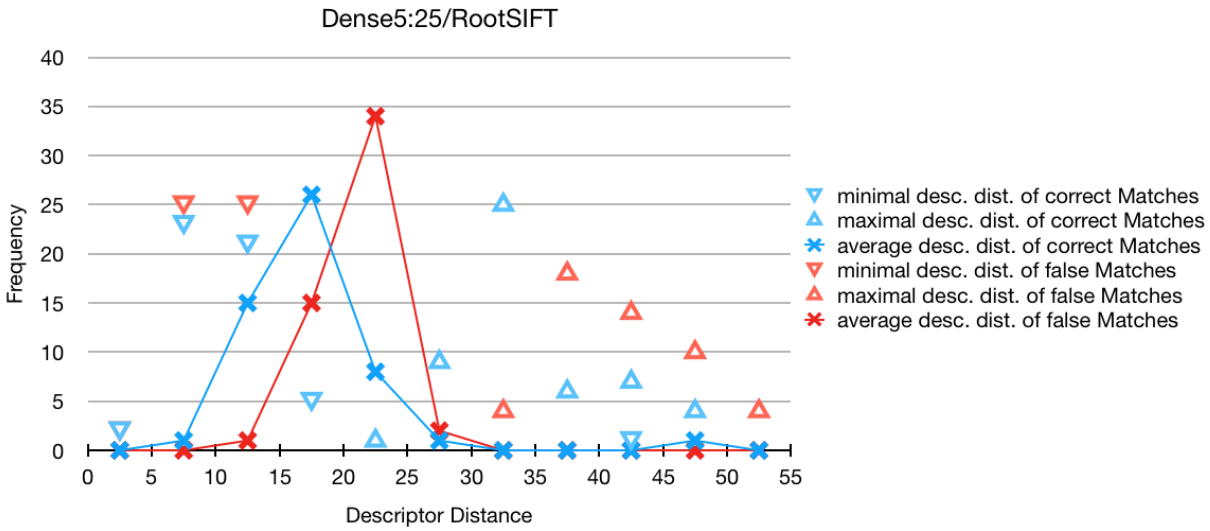


Figure 4.10: Illustration of the distribution of descriptor distance among correct and false matches for DENSE5:25/RootSIFT. Displayed are the frequencies of average descriptor distances for false and correct matches for each image pair of the dataset. Furthermore, the minimal and maximal descriptor distances of false and correct matches are depicted. The graph illustrates that the distance peaks for correct and false matches are not far apart from each other. Consequently, there is a great overlap between descriptor distances of false and correct matches. Thus, descriptor distance between correct and false matches can not be effectively utilized as an distinguishing criterion.

Maximal Descriptor Distance Threshold

Another approach to filter returned matches is the application of a maximal descriptor distance threshold. This method assumes that correct correspondences have a lower distance than false correspondences. So a maximal descriptor distance is defined and all returned matches featuring a distance above this are discarded. The choice of an appropriate threshold depends on the utilized descriptor as well as the similarity of the underlying image data. Thus, we compared the average distance of all correct matches against the average distance of all false matches for each of the top descriptors. This way we determined promising descriptor distance thresholds to evaluate. These are 25, 20, 15 and 10 for RootSIFT, 150, 130, 110 and 90 for LATCH64, 0.3, 0.25, 0.2 and 0.15 for U-SURF and 70, 60, 50, and 40 for BRIEF64, see also Figure 4.10.

The results of filtering matches with these maximal descriptor distances are displayed in Figure 4.11. As previously, for each combination the influence on average precision, the percentage of images containing at least 16 correct matches, as well as different pass rates requiring precision values of 1.0, 5.0 and 10.0 are shown. All pass rate values (yellow, orange and red lines) are capped by the percentage of image pairs containing at least 16 matches (green line).

All graphs show increasing average precision values as the allowed descriptor distance is decreased. Hence, in general more false than correct correspondences are eliminated, but also many correct correspondences are discarded as the drop in percentage of images containing at least 16 correct matches shows. Yet, compared to the ratio test the increases in precision are not that significant.

Overall, applying soft thresholds (2nd and 3rd column of each graph) does not eliminate too many correct correspondences. This means, there is no severe drop of the percentage of images containing at least 16 correspondences. However, the increase in precision is only moderate as

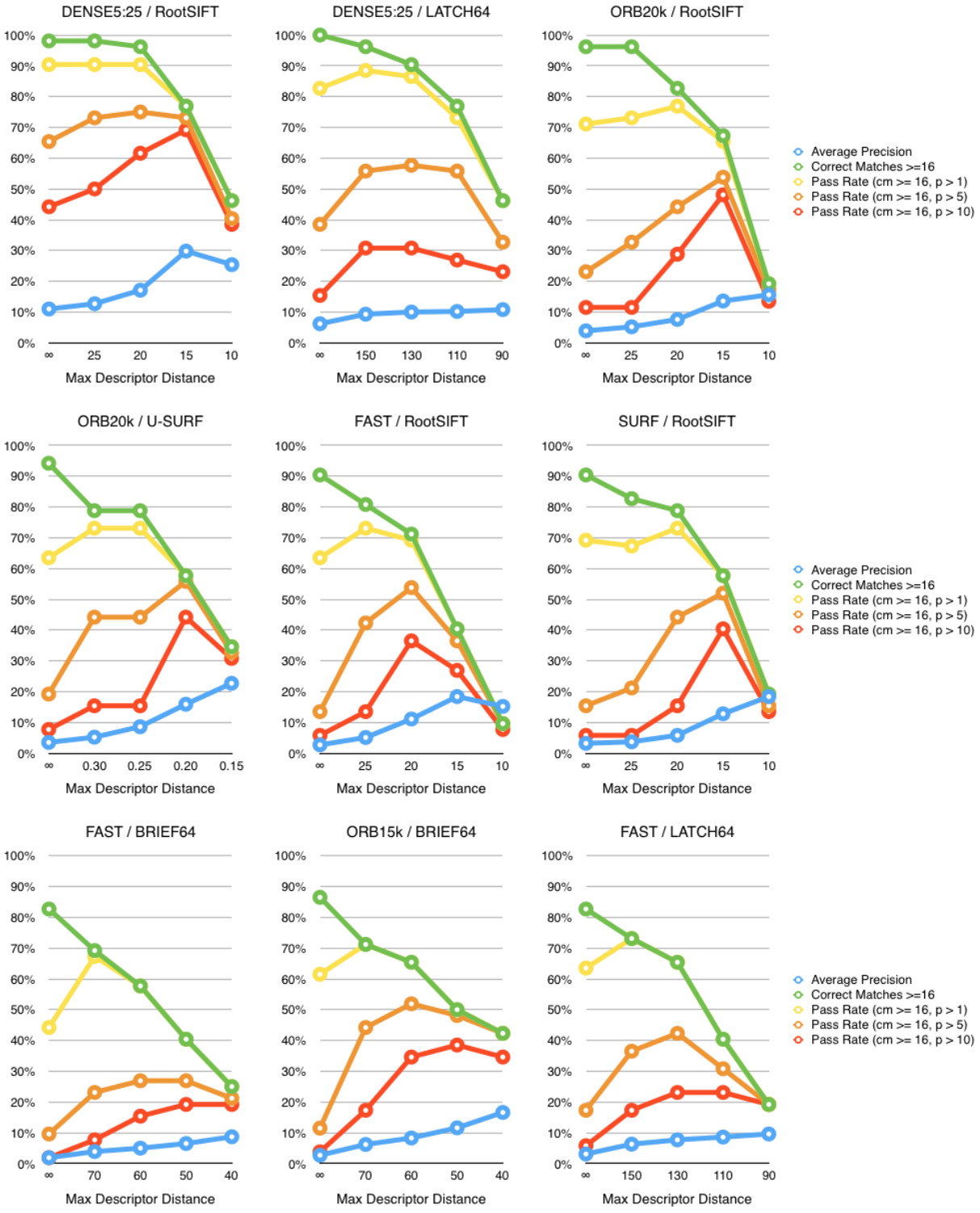


Figure 4.11: Results of applying a maximal descriptor distance threshold to the returned matches of the top 9 detector and descriptor combinations. For each the influence on average precision, the percentage of images containing at least 16 correct matches, as well as different pass rates are shown. All pass rate values (yellow, orange and red lines) are capped by the percentage of image pairs containing at least 16 matches (green line).

well, so that pass rates for precision values above 5.0 and 10.0 mostly remain below the results of applying the ratio test. On the other hand, stricter maximal descriptor distance values (4th and 5th column) eliminate very many correct correspondences. Thus, despite better average precision values, pass rates show no further increase, due to a lack of correct correspondences.

Again these results can be explained by the large distances correct correspondences show when comparing modern and historic images. Due to these the difference in average distance of correct and false correspondences is rather low and the overlap between correct and false correspondences in terms of distance threshold is high, see also Figure 4.10. Consequently, as soon as the threshold is raised many correct correspondences are eliminated as well, but if a low threshold is applied only few false correspondences are discarded.

Combining Ratio Test and Maximal Descriptor Distance Threshold

Since applying both, the ratio test as well as a maximal descriptor distance threshold, shows positive effects, but overall results are not satisfying, we additionally tried to combine both approaches. To do so, we apply a soft ratio which does not restrain the percentage of images containing at least 16 matches too much. Afterwards, we apply two different maximal descriptor distance thresholds for each detector and descriptor combination with the aim of discarding additional false matches. The results of this procedure are shown in Figure 4.12.

For comparison the left half of each graph (1st to 3rd column) depicts the results of exclusively filtering by maximal descriptor distance, while the right half (4th to 6th column) shows the results for applying the ratio test in combination with different maximal distance thresholds. In general, combining both approaches slightly improves results for average precision and pass rates requiring precision values above 5.0 and 10.0. However, using most of the top 9 combinations and applying both the ratio test as well as a maximal descriptor distance less than 60% of image pairs can be successfully registered, if we assume a need of at least 16 correct correspondences and precision values above 5.0.

Disparity Gradient Filter (DGF)

Our previous results show that commonly used match filtering approaches based on descriptor distance are not useful if matching historic and modern images. Due to the great changes present in these images even correct matches have large descriptor distances. Thus, descriptor distance is not a distinguishing feature between correct and false matches in rephotography. Another method for match filtering called Disparity Gradient Filter (DGF) has been successfully applied by Ali and Whitehead [2014]. This uses the measure of disparity gradient originally developed for stereo vision [Trivedi and Lloyd, 1985], whose application in general image registration has been proposed by Roth and Whitehead [2000]. The disparity gradient does not take photometric characteristics of keypoints such as descriptor distance into account. Instead, it is purely based on keypoint geometry.

Given an image pair and its correspondences the disparity gradient measures the geometric compatibility of two of these correspondences. Let P_1 and P_2 be keypoints in the first image and $f(P_1)$ and $f(P_2)$ be their correspondences in the second image, than their

$$\textit{disparity gradient } d = \frac{|(P_1 - P_2) - [f(P_1) - f(P_2)]|}{\frac{1}{2}|(P_1 - P_2) + [f(P_1) - f(P_2)]|}. \tag{4.1}$$

In case two correspondences conform well to each other their disparity gradient is small, while it increases as their conformity decreases. In general, we expect close keypoints to have similar

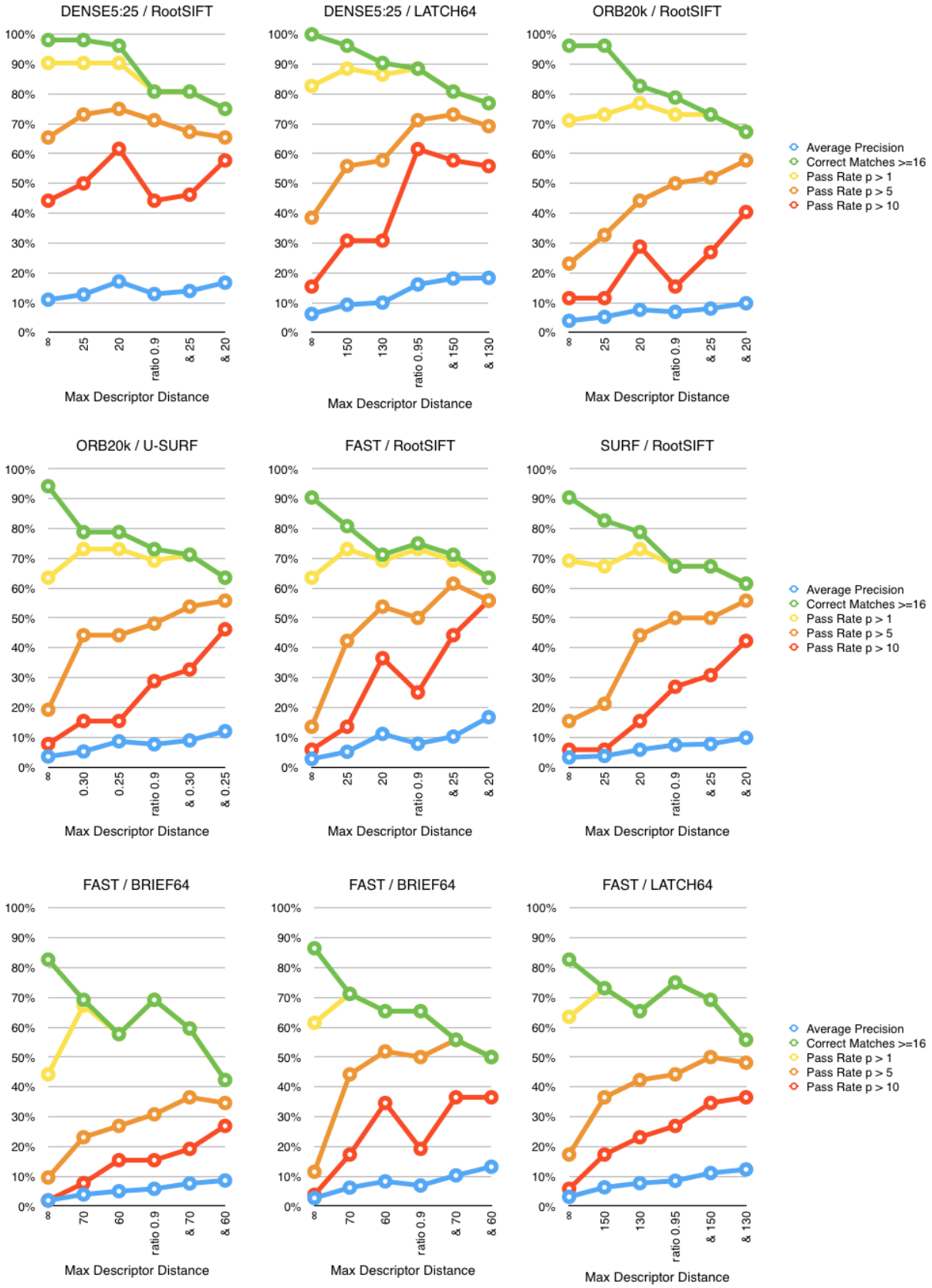


Figure 4.12: Results of applying the ratio test in combination with a maximal descriptor distance threshold. Combining both approaches slightly improves results for average precision and pass rate (right half of each graph).

projections. As a result, two correct correspondences have a small disparity gradient.

In stereo vision the inlier ratio is usually rather high and correspondences with a disparity gradient greater than 1.5 are rejected. Instead, in more general image registration the number of outliers increases. Thus, Roth and Whitehead [2000] propose a different approach for general image matching. First, the disparity gradient for each correspondence with respect to each other correspondence is summed up. Afterwards, correspondences with a high Disparity Gradient Sum (DGS) are removed. These two steps are repeated until the maximum DGS is less than twice the minimum DGS. Roth and Whitehead [2000] report that they are able to discard a significant number of false correspondences with this method. In general, at least 20% of correspondences are removed during application to images from a video sequence. Images of a video sequence recorded by a single camera during a short time span, feature much more inliers than the historic and modern image pairs of our dataset, which mostly have inlier rates of less than 5%.

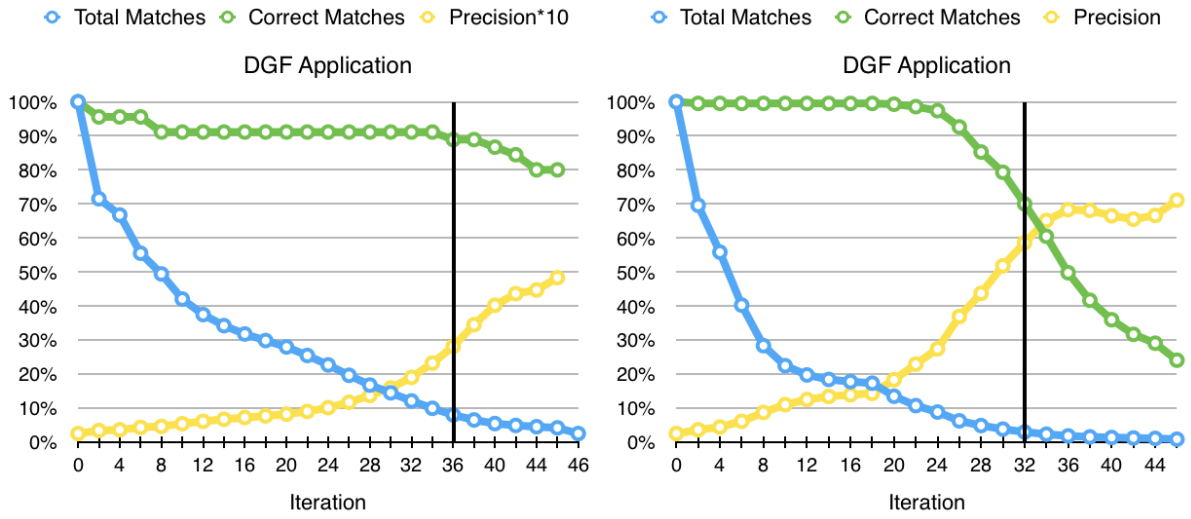
Ali and Whitehead [2014] applied DGF to image pairs of greater time spans and from different cameras. However, they provide no details on its performance or concerning its ability to handle small inlier ratios. Thus, in the following we evaluate the ability of DGF to cope with large amounts of outliers and determine its applicability to historic and modern image registration.

We follow the approach of Roth and Whitehead [2000], while we apply the following rule to discard false matches and a more relaxed termination criterion. During each iteration we determine the median DGS and discard all correspondences featuring a DGS greater than the median times a certain factor. Initially, this factor is set to $1^1/_2$. Yet, it decreases to $1^1/_4$, $1^1/_8$ and so on if $factor * median >= max\ DGS$ or less than 1% of correspondences were eliminated during the last iteration. During initial testing we realized that the termination criterion of $minimum\ DGS * 2 > maximum\ DGS$ is to tight for rephotographic image pairs. Thus, we changed it to $minimum\ DGS * 3 > maximum\ DGS$ after analyzing initial results.

Our evaluation reveals that the DGS is a very effective measure to distinguish correct from false correspondences. For almost every image pair DGF is able to discard a great number of false correspondences, while eliminating only few correct matches. This leads to a great boost in precision, even for image pairs with initial precision values below 1%.

The development of the remaining percentage of total and correct matches as well as precision during DGF application for two individual image pairs is depicted in Figure 4.13. For both image pairs features were detected with ORB20k and described with RootSIFT. Figure 4.13a displays a run where initial precision is only 0.25, while in Figure 4.13b initial precision is 2.5. For both image pairs more than 90% of total correspondences are rejected before termination, while only 10% to 30% of inliers are eliminated. In comparison the application of a rather soft ratio threshold of 0.9 discards approximately 90% (a) and 82% (b) of total correspondences for these two image pairs. Yet, at the same time 84% (a) and 71% (b) of correct correspondences are lost. The elimination of significantly more false than correct correspondences with DGF leads to a boost in precision. At termination precision raised to almost 3% for (a) which corresponds to a factor of 12, while for (b) it hits almost 60% corresponding to an increase of factor 24.

Looking at the results of individual image pairs we also noted that initial precision is not the dominant factor determining reachable precision. Instead, the initial number of correct correspondences is much more crucial. In the case of *United Nations 1* displayed in Figure 4.13b for instance the initial number of correct correspondences is 420 with a precision of 2.5, while for the image pair *Library* the precision is similar with 2.0 but only 177 correct correspondences were detected before applying any filter. At termination the precision of the first pair hits almost 60% corresponding to a factor of 24, while for the latter it can only be raised to 24% corresponding to a factor of 12.



(a) Bryanta

(b) United Nations 1

Figure 4.13: Detailed course of applying DGF to individual image pairs. Both graphs depict the development of the percentage of remaining total and correct matches as well as precision. The bold horizontal line depicts the iteration at which the algorithm stops with our termination criterion. Both graphs show the severe drop in total matches, while only few correct matches are discarded. This boosts precision. For (a) precision values increase from 0.25 to almost 3% at iteration 36 which corresponds to a factor of 12, while in the case of (b) precision raised from 2.5 to almost 60% at iteration 32 corresponding to a factor of 24. After our iteration point, precision values still increase, but at the same time the loss of correct matches becomes more significant.

In Figure 4.14 the results of applying DGF to the full dataset for all detector and descriptor combinations are presented. For all combinations DGF greatly outperforms all previously applied match filtering approaches. Furthermore, the percentage of images complying with a pass rate of precision values above 5.0 (orange line) is never capped by the percentage of images containing at least 16 correct correspondences (green line). This confirms that our termination criterion is not too tight.

Overall, after applying DGF the performance of classic detectors and descriptors on historic and modern images looks much more promising. In case we assume the requirement of at least 16 correct correspondences and precision values above 5.0 applying dense sampling almost 90% of the image pairs can be matched. For ORB/RootSIFT and SURF/RootSIFT approximately 80% of image pairs are matchable and for all other top combinations around 70%. In summary, DGF turns out to be an extremely efficient algorithm for match filtering. Furthermore, it is especially useful in matching images displaying great appearance changes, since it does not rely on the measure of keypoint distance as standard match filtering approaches do. Consequently, it is also useful for matching historic to modern images.

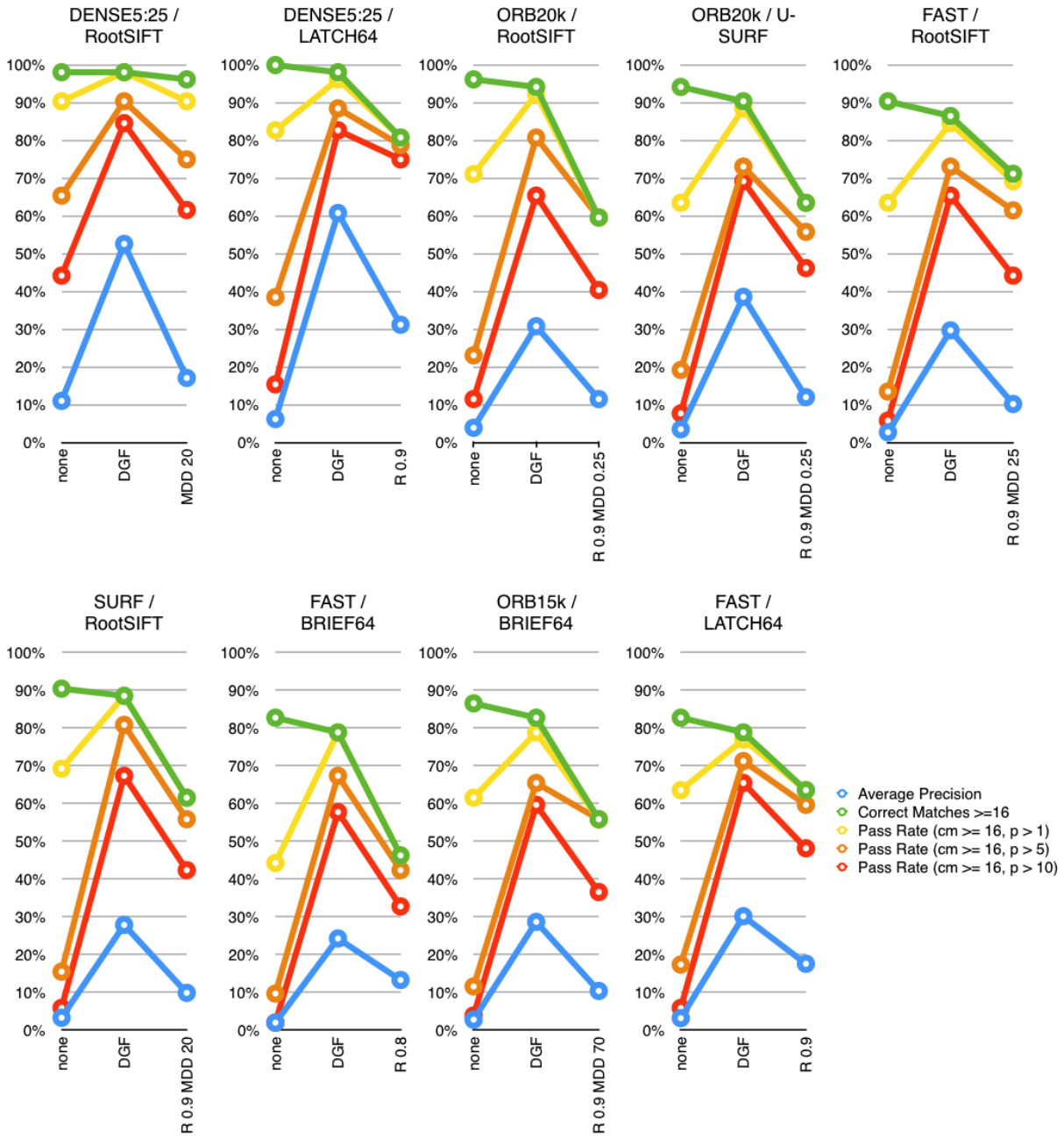


Figure 4.14: Results of applying the Disparity Gradient Filter (DGF) in comparison to previous filters. For each graph the first column shows the default performance of the respective combination. The second column displays the results after applying DGF, while the third row shows the results of the so far best performing match filter combination composed of the ratio test and a maximal descriptor distance threshold. For all feature combinations DGF greatly outperforms all previously applied match filtering approaches.

K-VLD

Another approach to filter returned matches, that uses additional constraints, beyond descriptor distance was proposed by Liu and Marlet [2012]. It is a semi-local matching method based on VLD and therefore called K-connected VLD-based matching (K-VLD)[5]. K-VLD is supposed to be able to estimate correct correspondences even if inlier rates drop below 5%. To do so it evaluates both geometric as well as photometric consistency in the local neighborhood of a keypoint pair.

Similar to DGF, K-VLD is based on the similarity between two corresponding pairs. It assumes that given two points P_1, P_2 and their correct correspondences $f(P_1), f(P_2)$ the photometric information around lines (P_1, P_2) and $(f(P_1), f(P_2))$ is similar. After checking geometric consistency of two correspondences, a SIFT-like descriptor is used to describe the image area around the lines (P_1, P_2) and $(f(P_1), f(P_2))$. Based on these descriptions the gVLD-consistency is computed. This assess the similarity between both lines including geometric as well as photometric consistency. K-VLD uses this gVLD-consistency to compute the consistency of a correspondence with its neighboring correspondences. It requires each correspondence to have at least k gVLD-consistent neighbors to be classified as an inlier. Besides, an additional constraint on the proportion of geometry-consistent neighbors is used to discard remaining outliers.

At first we apply K-VLD to all returned matches. Results show that compared to the ratio test and maximal descriptor distance threshold, K-VLD is much more efficient in filtering false correspondences. Its advantage is, that, similar to DGF, it does not rely on keypoint descriptor distances, which are less significant in the case of major appearance changes between images. However, compared to DGF, K-VLD discards a larger number of correct correspondences. We believe this results from its reliance on photometric similarity of lines between two correspondences. Due to the great amount of occlusion in modern and historic image pairs this may differ greatly even among correct correspondences.

As a consequence, the success of the K-VLD filter relies not only on the initial ratio of correct correspondences, but also the total number of correct correspondences. This is illustrated in Figure 4.15, which displays K-VLD results arranged by initial precision and number of correct matches. Especially for initial precision values below 1.0 failure cases accumulate as the initial number of correct matches decreases. In these cases K-VLD commonly discards to many correct matches. Note that in failure cases K-VLD often terminates with zero matches left, while only in individual cases a set of false positives is selected or the inlier ratio among the returned correspondences is less than 5.0.

Considering the whole dataset K-VLD performs a little poorer than DGF in terms of pass rate, see Figure 4.16. Furthermore, the drop in percentage of image pairs containing at least 16 correct matches (green line) nicely illustrates K-VLD's tendency to eliminate more correct matches. Yet, K-VLD reaches similar increases in average precision. So in terms of increasing precision for single image pairs it outperforms DGF.

Thus, additionally we combined both filters. The idea is to first use DGF to eliminate as many outliers as possible, while discarding only few correct matches. Afterwards, K-VLD is applied to further increase precision, especially in those cases were DGF terminates with precision values below 10.0. As the results in Figure 4.16 display the combination of K-VLD and DGF with the previous termination factor of 3.0 is most efficient. On the other hand, relaxing the termination factor to 3.5 does not result in an significant increase of image pairs featuring precision values above 10.0 and results in a decrease of average precision.

[5]code at: `https://github.com/Zhe-LIU-Imagine/KVLD` (accessed January 30th, 2016)

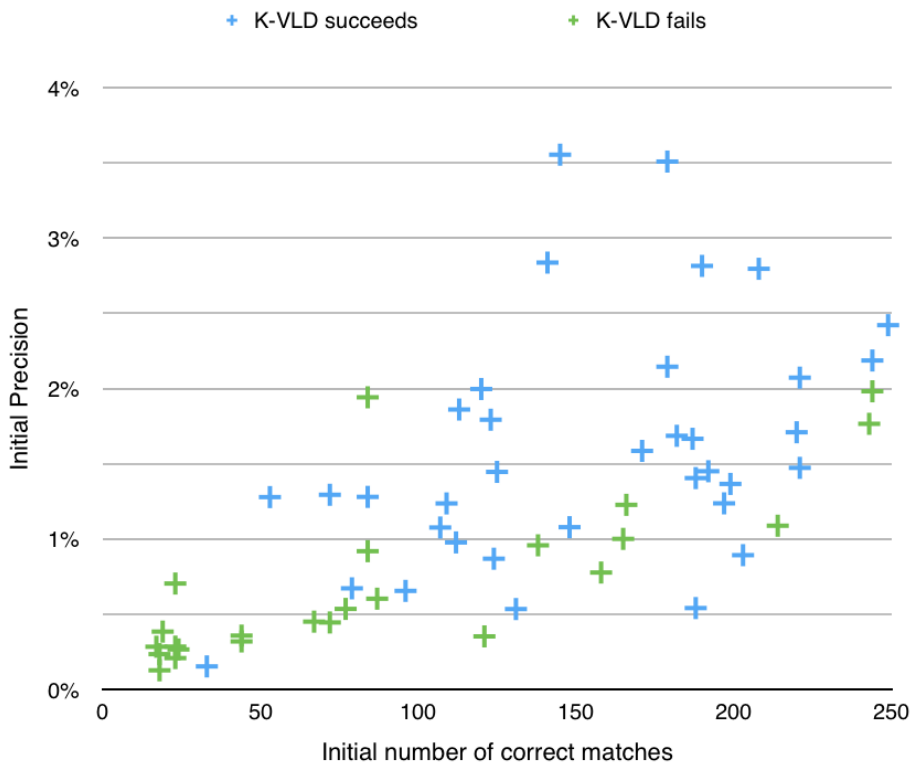


Figure 4.15: Illustration of K-VLD filter results arranged by initial number of correct matches and precision. Displayed are the results for the top as well as the bottom two detector and descriptor combinations, namely DENSE5:25/RootSIFT, ORB15k/BRIEF64, FAST/LATCH64. Only image pairs with initial precision values below 4 and an initial number of correct matches between 15 and 250 are displayed. The application of K-VLD is regarded successful if more than 16 correct correspondences remain after filtering and the precision value is at least 5.0. The graph illustrates, that initial precision is not the only factor determining the success of K-VLD. Instead, the initial number of correct matches is relevant as well. Especially for initial precision values below 1.0 failure cases accumulate as the initial number of correct matches decreases.

4.3.5 C: Final Alignment Computation

In the last section we evaluated different methods of correspondence filtering to increase precision. Yet, the final goal in rephotography is to determine the parameters of the transformation that allows to align both images. In our case this is a homography matrix that provides a good global alignment of both images. Since the images of the dataset do not show single plains, they can not be perfectly aligned via a homographic transformation. However, as the dataset mainly contains images that were already well aligned at the beginning, only few image regions in the background or at the image sides do not match after transform application.

After applying DGF only or combining it with K-VLD, for all detector and descriptor combinations 10% to 30% of image pairs still feature precision values below 10.0, see Figure 4.16. This means we still face rather low inlier rates of less than 10% for a significant amount of image pairs. To be able to match these pairs, we require a method that is very robust and able to handle high outlier ratios.

In search of a global image transformation such as a homographic transformation RANSAC [Fischler and Bolles, 1981] is a popular method to distinguish inliers and outliers. It randomly selects a minimal subset of correspondences to estimate a possible transformation matrix and

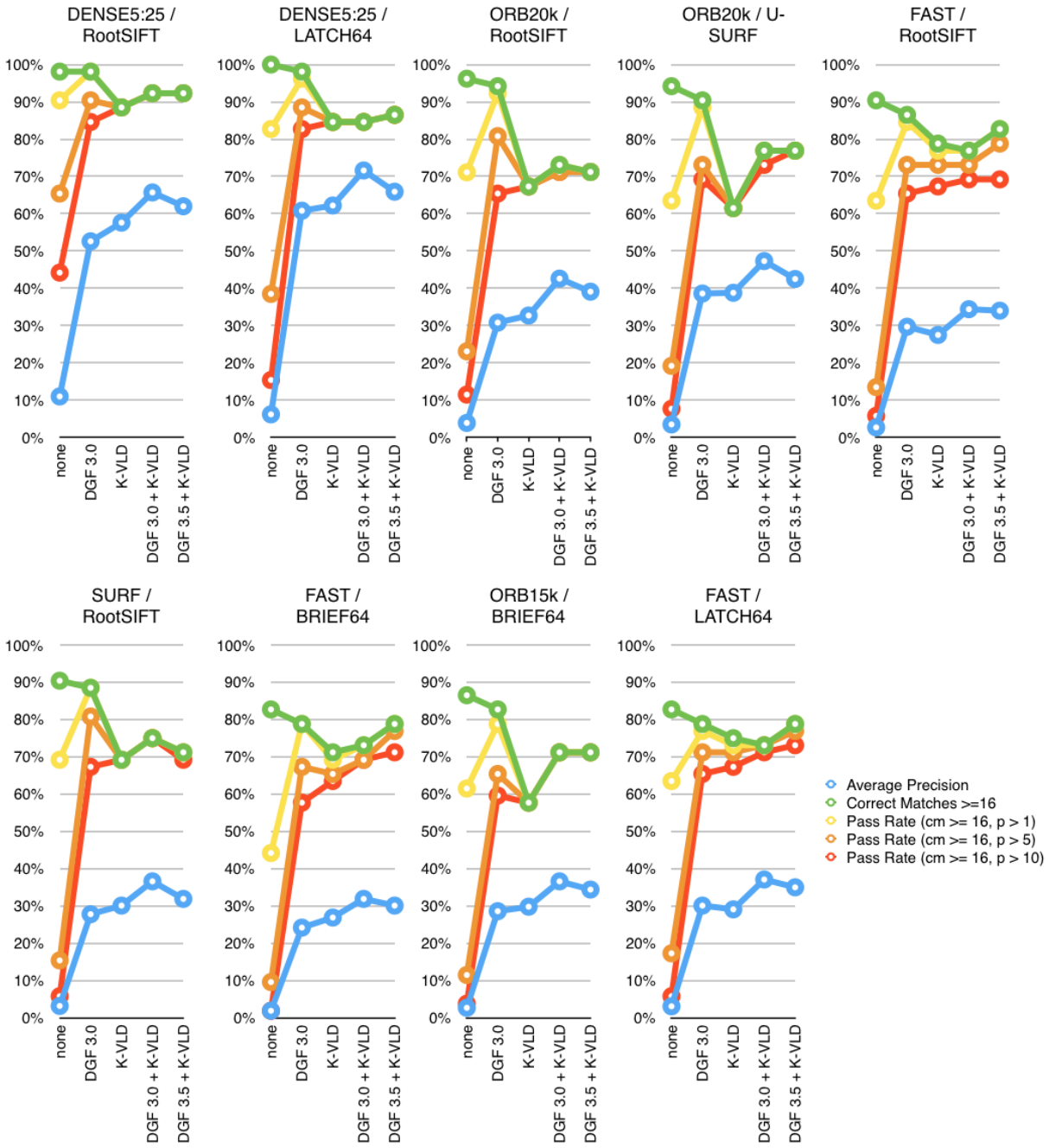


Figure 4.16: Results of applying K-VLD in comparison to DGF. For most detector and descriptor combinations pure K-VLD performs a little poorer than DGF in terms of pass rates with precision values of 1.0 and 5.0 (yellow and orange line), while reached average precision values are almost equal (blue line). Finally applying DGF with a termination factor of 3.0 followed by K-VLD (4th column), shows the best performance for most combinations.

ranks such by the total number of correspondences compatible with it. The number of necessary iterations k (samples to be drawn) to determine the correct homography at a certain probability p can be estimated by the following formula:

$$k = \frac{log(1-p)}{log(1-w^n)} \tag{4.2}$$

Here n is the number of points required to estimate the model and w depicts the inlier rate of the provided data.

In case of a homographic transformation n equals 4. If the desired probability to find the correct solution is 0.95 and the rate of correct correspondences is 10%, 29995.8 iterations are required. In case the inlier rate drops below 10% for instance to 0.05, up to 479000 iterations are required. On the other hand, if we desire to estimate a fundamental matrix requiring 8 correspondences an inlier rate of 0.2 demands 1.17 *million* iterations, while only for inlier rates of 0.45 the requested number of iterations decreases below 2000.

PROSAC [Chum and Matas, 2005], an extension of RANSAC, is able to reduce the number of necessary iterations due to advanced sample selection techniques. Nevertheless, it is not well suited for image pairs with precision values below 50%. The same applies to other approaches from statistics such as Least Median of Squares (LMedS) [Rousseeuw, 1984]. ORSA [Moisan and Stival, 2004] is one of the very few methods that is able to handle outlier rates around 90%. In fact, the authors claim that an inlier rate of 0.07 is sufficient [Moisan et al., 2012].

Since rephotographic image pairs suffer from high outlier ratios we analyze the success of ORSA for homography estimation and compare it to standard methods including RANSAC [Fischler and Bolles, 1981] and LMedS [Rousseeuw, 1984]. In the following we first present our method to estimate the quality of an automatically estimated homography. Afterwards, follow the results of the different model estimation algorithms for our dataset as well as a detailed analysis of cases where homography estimation fails.

Assessing the Quality of Estimated Homographies

Unfortunately there is no well-established standard, how to evaluate the quality of an automatically estimated homography. Instead, related studies evaluating the performance of feature detectors and descriptors only compare these based on recall and precision [Hauagge and Snavely, 2012; Mikolajczyk and Schmid, 2005]. Yet, the minimal number of correct matches and whether this is high enough for homography or fundamental matrix estimation remains unconsidered. At least the evaluation by pass rate [Gat et al., 2011] includes the required minimal number of correct matches, but their quality is not assessed. Correct correspondences might have a low quality if they are limited to a single image region. In this case estimated homographies will be overfitted to this region, which results in great distortion of distant areas. For fundamental matrix estimation, it is also a sign of low quality if all correct correspondences lie on a single plane.

The modern and historical image pairs of our dataset often suffer from high occlusion due to structural change. Consequently, correct correspondences are already limited to certain regions due to image composition. Furthermore, these regions show great differences in appearance. Thus, it is likely that in the end all correct correspondences established by a certain detector and descriptor combination are limited to a single region. Hence, especially in the case of matching images suffering from high occlusion and great appearance changes it is necessary to assess the quality of correct correspondences in terms of the desired model estimation.

(a) Good Alignment **(b)** Bad Alignment

Figure 4.17: Examples of (a) good and (b) bad alignment of an image pair after transformation by an automatically estimated homography.

In the context of 3D reconstruction it is common practice to work with a looping sequence of images providing high quality ground truth data. An example is Strecha's castle dataset [Strecha et al., 2008] which provides ground truth for internal and external camera parameters. In this case camera rotation and translation information can be extracted from estimated fundamental matrices and compared to ground truth. This allows to measure and compare distance to original viewpoint [Bae et al., 2010] and angle error or error accumulation across the whole sequence [Liu and Marlet, 2012]. Yet, collecting such high quality 3D data is time consuming. Especially if it is required for more than a single location.

Thus, for the image pairs of our dataset we would like to evaluate the estimated homography based on the ground truth homography established by manual point selection. A common strategy for assessing the quality of a homography is via the reprojection error of all selected inliers [Moisan and Stival, 2004]. This is also the criterion most algorithms aim to minimize during homography estimation. However, the reprojection error, does not provide any information on the quality of the set of inliers selected. Hence, it is possible to choose a set of false correspondences that nonetheless has a low reprojection error. Alternatively the presence of correct correspondences (evaluated by the manual homography) among the set of selected inliers can be checked. Yet, it is difficult to estimate the necessary overlap between these two sets required for good homography estimation. In some cases false matches hardly influence homography computation if they are close enough to the correct ones. Especially after application of DGF and K-VLD this is often the case. On the other hand, a very small subset of all correct correspondences can be sufficient for model estimation, if it represents the global perspective change. Yet, if such a small set is limited to a single image region no acceptable homography can be computed.

Our Approach

Instead, we assess the estimated homography purely by its similarity to the manually estimated ground truth homography. For similarity computation we project all image corners together with the image center by both homographies and calculate the mean distance of all projected pairs. Furthermore, we compute the surface area between all projected corners and determine the factor by which the automatic and manual projections vary.

To determine which mean distance and area deviations are acceptable upon automatic homography estimation, we manually evaluate edge overlays constructed from the estimated transformation. For an example of an acceptable and non acceptable result refer to Figure 4.17.

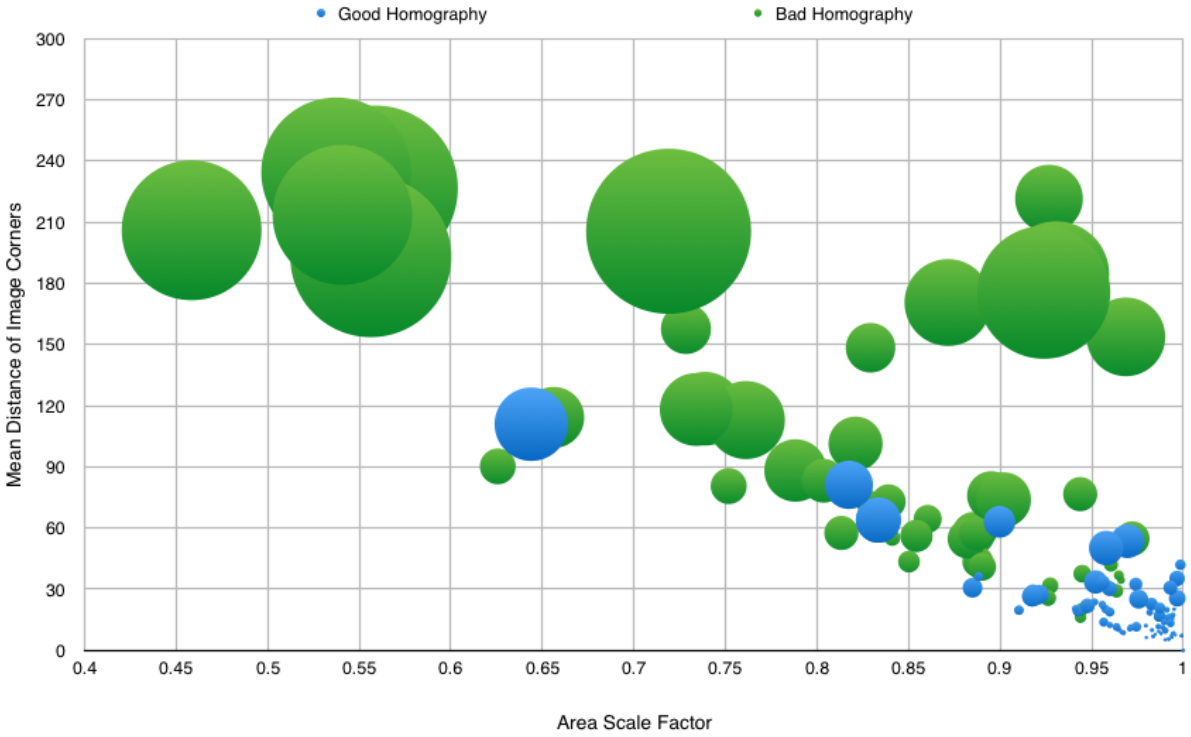


Figure 4.18: Distribution of good and bad estimated homographies based on our proposed evaluation criteria, including mean distance of image corners and the area scale factor between warped images. The graph nicely depicts that image pairs with a mean distance of image corners below 30 and an area scale factor above 0.95 are correctly aligned (strong criterion). Furthermore, most of the image pairs with a mean corner distance below 60 and an area scale factor of at least 0.9 are well aligned (weak criterion). Each dots thickness represents the number of inliers used for homography estimation.

Overall, we manually analyzed the results for 156 image pairs. In detail these include all pairs of the dataset aligned by using ORB15k/BRIEF64 and DENSE5:25/LATCH with DGF followed by K-VLD, plus all pairs aligned by DENSE5:25/LATCH with DGF. For all combinations LMEDS was used for homography estimation. This way we receive a sufficient amount of well and not well aligned image pairs.

Figure 4.18 illustrates the distribution of good and bad generated alignments based on the mentioned evaluation criteria, including mean distance of image corners and the area scale factor between warped images. To receive scale factors between 0 and 1, we always divide the smaller by the larger area. Image pairs with a mean distance of corner and center points above 250 are not displayed. Figure 4.18 depicts that image pairs with a mean distance of image corners below 30 and an area scale factor above 0.95 are correctly aligned (bottom right square). Besides, most of the image pairs with a mean corner distance below 60 and an area scale factor of at least 0.9 are well aligned. Hence, we use these boundaries as weak (mean distance < 60, area scale factor > 0.9) and strong (mean distance < 30, area scale factor > 0.95) criteria to judge whether an image pair has been successfully aligned via an estimated homography matrix.

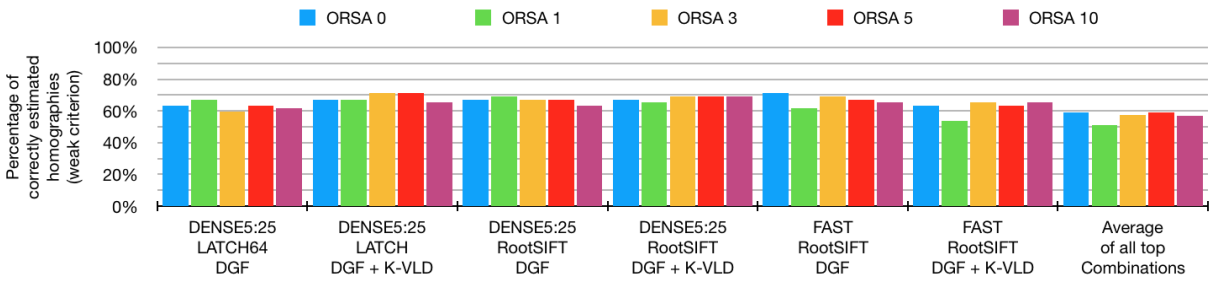


Figure 4.19: Comparison of different maximum error values for ORSA. The graph displays the percentage of good homographies, by the weak criterion (mean distance < 60, scale factor > 0.9), for individual matching pipelines as well as the average of all top combinations. Overall, results are unevenly distributed and there is no explicit best choice of error value. However, on average ORSA 0 and 5 outperform other maximal error thresholds.

Methods for Model Estimation

ORSA is one of few methods able to handle outlier rates around 90% [Moisan et al., 2012]. In general, it does not require an initial reprojection error threshold, that distinguishes model inliers from outliers. Instead, it automatically estimates the error threshold, called precision in the context of ORSA. Yet, there is the option to define a maximum error. In this case ORSA still determines the final precision automatically, but it will always be below the defined maximum error. Setting a maximum error may increase final model precision and prevent ORSA from getting caught in a local minimum. Thus, we evaluated the performance of ORSA without setting any precision (ORSA 0), as well as its performance with maximal error set to 10, 5, 3 and 1.

Overall, we were not able to detect significant differences in performance for setting none or the applied maximal precision values, see also Figure 4.19. For many detector and descriptor combinations, performance dropped for maximal error values of 1, 3 or 10, while for few one of them showed slightly increased performance compared to ORSA 5 and 0, which on average show the best performance. Since the image pairs do not change, we expect the maximal error to be independent of the applied detector and descriptor combination. Hence, we attribute the small variations for certain combinations to chance and in the following only the results for ORSA 5 are presented.

RANSAC [Fischler and Bolles, 1981] is a very prominent algorithm used for model estimation in the presence of outliers. It selects an optimal model based on inlier count. Consequently, it requires an error threshold to distinguish inliers from outliers.

Thus, an appropriate error threshold for RANSAC needs to be chosen. In order to do so we tested thresholds 1, 3, 5 and 10. Similar to the results for ORSA, there was no single threshold that showed the best performance across all detector and descriptor combinations, see figure 4.20. Instead, for most combinations a threshold of 5 provides good results, but in some cases other thresholds show slightly improved performances. Especially on average there is no clear best choice of error value.

Additionally, we analyzed the count of inliers selected by RANSAC compared to the number of correct matches according to the manually based homography. Results for DENSE5:25/LATCH with DGF and K-VLD application are displayed in Figure 4.21. The graph illustrates that for very high inlier ratios above 90%, all methods select approximately correct inlier numbers. Yet, especially in the mid range with precision values of 30% to 80%, only an error threshold of 5 leads to an inlier selection resembling the number of correct matches. A stronger threshold of

3 results in overfitting while a threshold of 10 seems to weak. Comparisons for other detector and descriptor combinations look similar. Hence, we only present the results of RANSAC with an error threshold of 5 in the following.

LMedS [Rousseeuw, 1984] ranks estimated homography models by the median re-projection error of all correspondences. As a result, it does not depend on an error threshold but theoretically requires more than 50% of inliers.

Optimistic Baseline In order to determine, whether bad homography estimation results from the model estimation algorithm or is caused by a poor correspondence quality, we additionally compute homographies based on the whole subset of correct matches. This way we can identify image pairs, where correct correspondences are clustered at few image regions, which do not adequately represent the global image transformation.

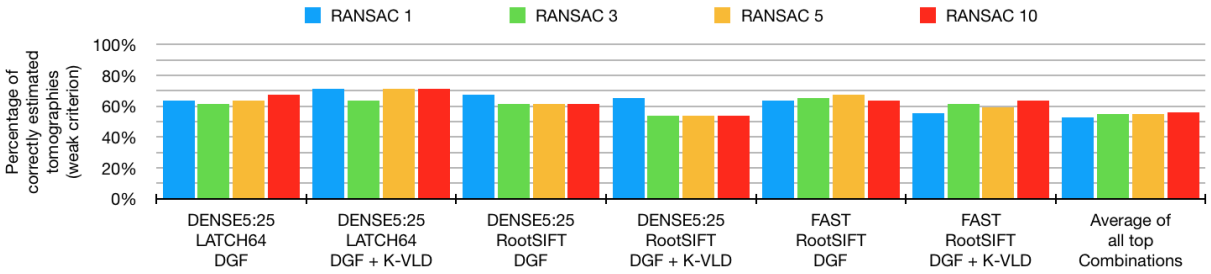


Figure 4.20: Comparison of different error thresholds for RANSAC. The graph displays the percentage of good homographies, by the weak criterion (mean distance < 60, scale factor > 0.9), for individual matching pipelines as well as the average of all top combinations. Results are unevenly distributed and there is no clear best choice of error threshold, even on average there is no significant difference between the results of different error thresholds.

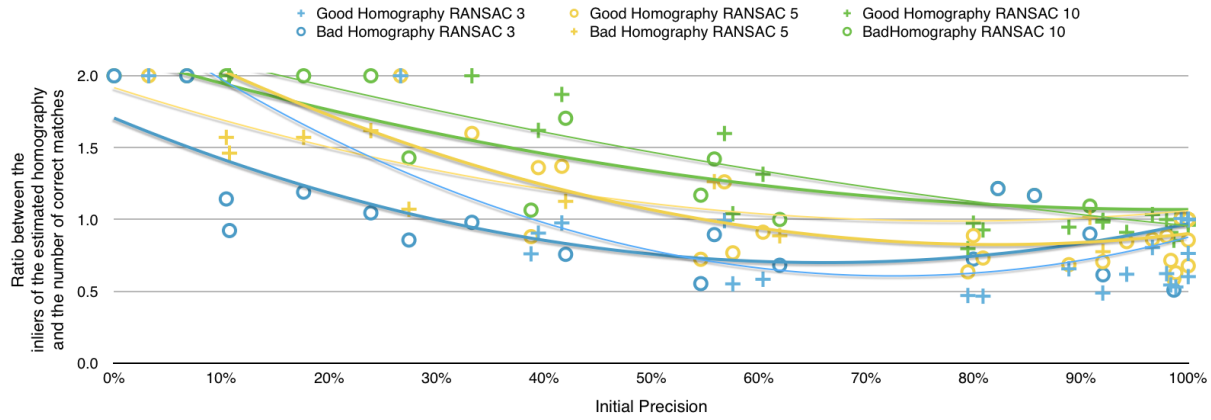


Figure 4.21: Comparison of different error thresholds for homography estimation via RANSAC. The graph displays results for DENSE5:25/LATCH64 with DGF and K-VLD. Individual image pairs are distributed based on the percentage of inliers (x-axis) as well as the number of inliers of the estimated model compared to the number of correct matches (y-axis). A y value above 1.0 indicates that more inliers are selected than correct matches exist, while a value below 1.0 suggests that only a subset of the correct matches has been selected. Lines represent polynomial functions of order 2 fitted to the data. Thick lines belong to badly estimated homographies while thin lines show the trend for good homography estimation results.

Results

Figure 4.22 displays the results of applying ORSA 5, RANSAC 5 and LMEDS for homography estimation after applying DGF exclusively and followed by K-VLD for all different detector and descriptor combinations. For comparison pass rate 16, with a required precision of 10%, is depicted as well as results for optimistic baseline.

With few exceptions the percentage of image pairs that can be successfully registered remains far below pass rate for all model estimation methods. For most detector and descriptor combinations pass rates reach values around 70% after applying DGF combined with K-VLD, while for DENSE5:25/RootSIFT a pass rate of 90% and for DENSE5:25/LATCH a pass rate above 80% is reached, remember Figure 4.16. Instead, correct model estimation ranges from 40% to maximally 70%. Overall, there is a significant lack between expected and actual performance of model estimation. In the following this is examined in more detail.

At first, we compare pass rate and optimistic baseline. If DGF and K-VLD are applied optimistic baseline is usually comparable or a little lower than pass rate. This indicates that in general the correct matches remaining after filter application properly represent the global image transformation. An exception is the combination of DENSE5:25/RootSIFT which shows very low values for optimistic baseline in comparison to pass rate. This suggests, that for densely sampled RootSIFT applying K-VLD is to restrictive and correct matches are clustered at few image regions, which leads to false homography estimation, see Figure 4.23. For other combinations this is only a minor problem of individual image pairs. If only DGF is applied results for optimistic baseline are usually higher than for pass rate. This is, since pass rate requires a precision of 10%, while homography computation may succeed as soon as four correct matches exist, which are well distributed across the entire image. As DGF keeps more correct matches at a loss of precision, it aids optimistic baseline computation, and such becomes over optimistic. Consequently, in the case of DGF application it makes more sense to compare model estimation results to pass rate. On the other hand, taking the results of optimistic baseline into account, the lack between pass rate and successful model estimation can only be a result of low precision values. This means 10% of inliers are not enough for all model estimation algorithms.

Comparing the results of LMEDS, ORSA 5 and RANSAC 5, not many differences can be observed. All of them show equal performances or LMedS performs a little better. Only for few combinations, including DENSE5:25/RootSIFT and ORB15k/BRIEF with DGF and K-VLD application, ORSA 5 outperforms both other methods. Considering that ORSA is supposed to be able to cope with precision values around 10% [Moisan et al., 2012], while LMedS theoretically requires much higher precision, these results are rather surprising. Consequently, in the following we will take a closer look at the results of individual image pairs.

Analysis of Failure Cases

To analyze the main cause of failure of homography estimation we looked at the individual results of DENSE5:25/RootSIFT and DENSE5:25/LATCH, which have the most promising pass rates of ≈90% and ≈85%. Figure 4.24 illustrates examples of good and bad homography estimation for DENSE5:25/LATCH after applying DGF and K-VLD. The graph allows a more detailed comparison of all three model estimation algorithms.

In general, homography estimation suffers from low precision values. In case the inlier rate is below 30% model estimation fails for all methods and images with only rare exceptions. Yet, interestingly ORSA also estimates a bad model for a substantial number of image pairs with precision values above 80%. In these cases often a subset of only 5 matches is selected, that

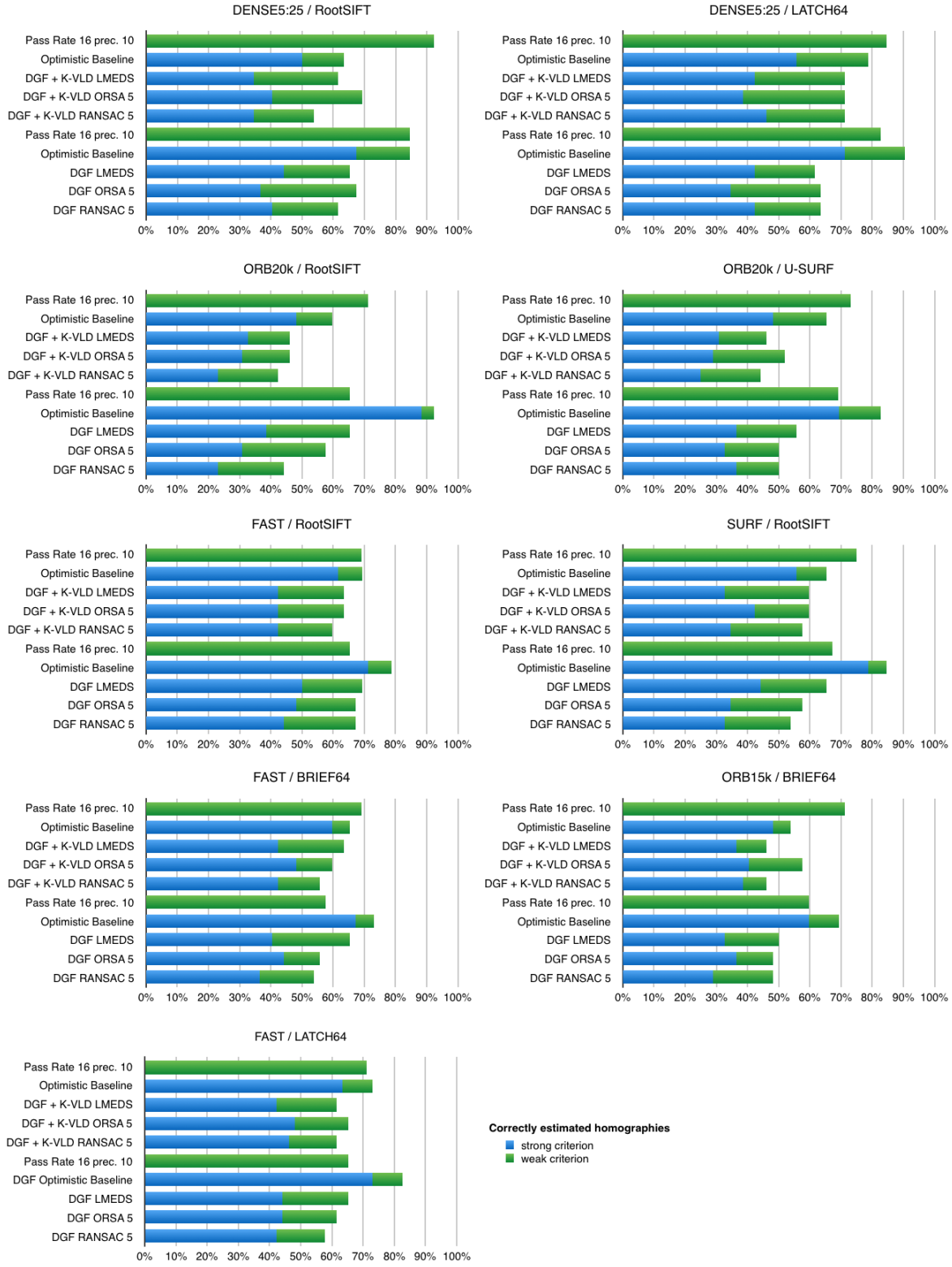


Figure 4.22: Comparison of different homography estimation algorithms across all detector and descriptor combinations. The blue part of each bar depicts the percentage of images for which homography estimation succeeded according to the strong criterion (mD<30 and sF>0.95), while the whole bar including the green part represents the success of homography estimation according to the weak criterion (mD<60 and sF>0.9). For comparison pass rate for at least 16 correct matches and precision values above 10% is displayed.

Figure 4.23: Illustration of remaining correct matches for DENSE5:25/RootSIFT after application of DGF and K-VLD. At the top two image pairs are shown, whose remaining correct matches are clustered at a single small image region. This leads to bad homography estimation, since the estimated homography does not adequately present the global image transformation. An overlay of the image alignment proposed by homography estimation from all correct matches is displayed on the right. Instead, in the bottom example, correct matches are well distributed across the entire image, leading to the desired homography estimation.

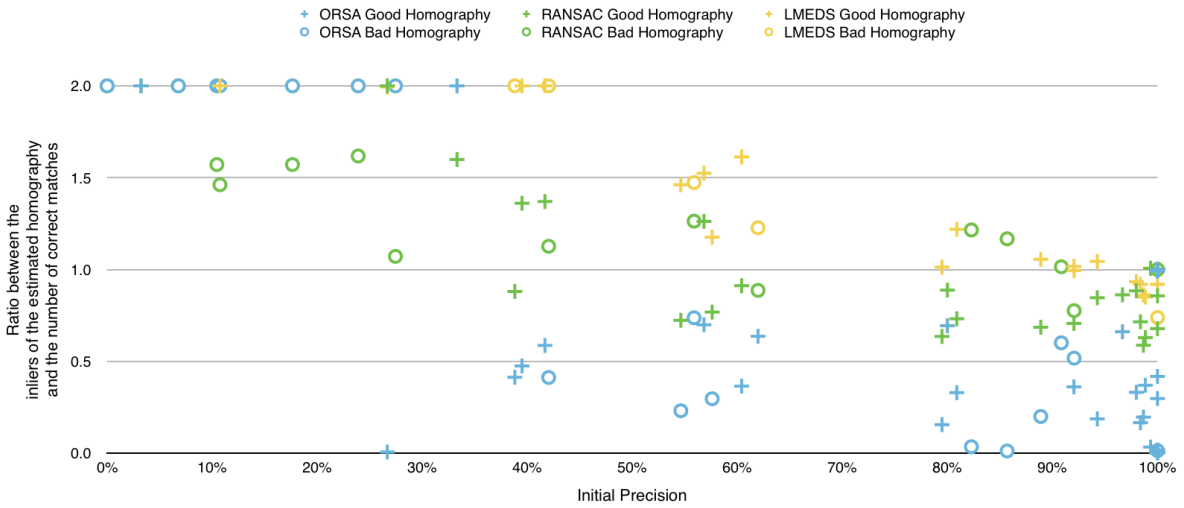


Figure 4.24: Comparison of different robust algorithms for homography estimation including RANSAC and ORSA with an error threshold of 5 and LMedS. The graph displays results for DENSE5:25/LATCH64 with DGF and K-VLD. Individual image pairs are distributed based on the percentage of inliers (x-axis) as well as the number of inliers of the estimated model compared to the number of correct matches (y-axis). A y value above 1.0 indicates that more inliers than correct matches exist are selected, while a value below 1.0 suggests that only a subset of the correct matches has been selected.

have a reprojection error close to 0, but do not represent the global transformation. For ORSA estimated homographies often feature residual error significantly below 1 and, compared to the number of correct matches, only very few inliers are selected for model estimation. This explains why varying the maximal error shows hardly any effect. LMedS usually fails for less than 50% of inliers, but for high precision values its success is very reliable compared to that of ORSA and RANSAC. RANSAC on the other hand, succeeds in aligning some images with lower precision values balancing performance shortfalls among higher inlier rates.

4.3.6 Robustness to Scale and View Changes

Previously we analyzed the quality of homography estimation for historic to modern image pairs, which can be used for post-processing rephotographs. However, we are also interested in guiding the process of retaking a photograph as proposed by Bae et al. [2010]. For this the goal is the recovery of the 6DoF camera pose of the historic image, which requires at least 8 correct correspondences. Furthermore, upon location recovery, larger changes in viewpoint have to be expected. Thus, in the following we analyze the ability of our established detector, descriptor and match filtering pipeline in the context of scale and perspective change.

Scale Change

To simulate a scale change for the image pairs of our dataset, we apply a perspective transformation to all modern images, using the following transformation matrix:

$$H_s = \begin{pmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{4.3}$$

with $s \in \{0.7, 0.85, 1.0, 1.15, 1.3\}$. Consequently, we evaluate matching performance up to a difference in scaling of one third of the original image size. Greater variations in image scale are unlikely in the context of rephotography.

Figure 4.25 illustrates the effects of scale change on pass rate and homography estimation for DENSE5:25/LATCH after DGF and K-VLD application. The graphs show results for pass rate 16 with precision above 10% and 50% and results of homography estimation for all three model estimation algorithms. Compared to our previous evaluations, we decided to display results for the high precision threshold of 50%, since this more adequately resembles the results of homography estimation. All three graphs illustrate, the larger the difference in scale, the larger

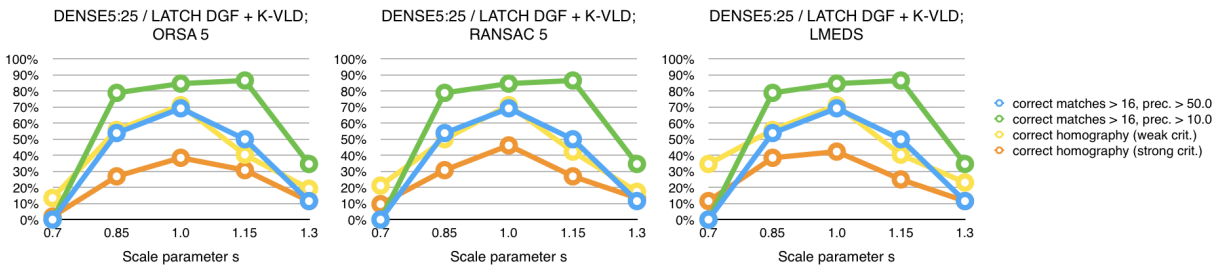


Figure 4.25: Effects of scale change on pass rate and homography estimation for DENSE5:25/LATCH with DGF and K-VLD. For each scale pass rate 16 with required precision above 10% and 50% as well as the percentage of good homographies according to the weak and strong criterion is displayed.

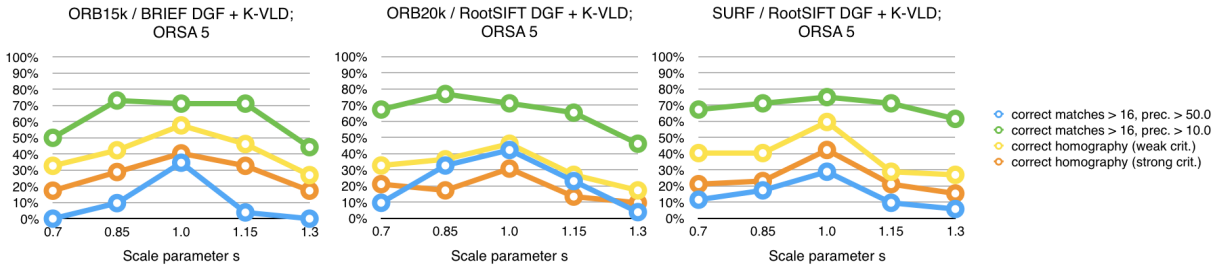


Figure 4.26: Effects of scale change on pass rate and homography estimation for top combinations including the local feature detectors ORB and SURF detected at different scales. For each scale pass rate 16 with required precision above 10% and 50% as well as the percentage of good homographies according to the weak and strong criterion is displayed.

the drop in performance in terms of pass rate as well as successful homography estimation. This is independent of the model estimation technique used.

These results are not surprising in the context of dense sampling, since all keypoints are sampled at a single scale. Consequently, if image scale varies, simultaneously the content of regions covered by a single keypoint varies and it becomes more difficult to match the descriptors of these different keypoint regions. Thus, if a matching pipeline including dense sampling is used, there should only be small variations in scale. Scale changes in the range of 15% are acceptable.

Figure 4.26 depicts the effects of scale change for combinations using the local feature detectors ORB and SURF, which are detected at different scales. The graph shows results for the combinations ORB15k/BRIEF, ORB20k/RootSIFT and SURF/RootSIFT with DGF and K-VLD application and model estimation with ORSA 5. As previously the results for pass rate and homography estimation are shown. All three graphs illustrate, compared to dense sampling, for local feature detectors the drop in performance due to scale change is only weak. Especially with regard to large scale changes of approximately one third of the original image size, local feature detectors outperform densely sampled features. Hence, if large scale changes are an issue of the underlying data, as in cases where historic images are sorted or aligned to a city model [Schindler and Dellaert, 2012], instead of dense sampling, a large amount of locally detected features should be applied.

Figure 4.27: Illustration of the perspective change enforced on each image pair.

Perspective Change

To simulate perspective change, the following transformation matrix is used:

$$H_p = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ p & 0 & 1 \end{pmatrix} \tag{4.4}$$

with $p \in \{-0.0005, -0.00025, 1.0, 0.00025, 0.0005\}$. For a visualization of the effect this has on a single image refer to Figure 4.27. Since the images do not consist of a single plane, the perspective transformation we apply creates an artificial view, which could not be captured at the real scene. Yet, using this artificial view we are able to approximate the effects of perspective change without collecting additional image material.

Figure 4.28 depicts the effects of perspective change on pass rate and homography estimation for DENSE5:25/LATCH after DGF and K-VLD application for all three model estimation algorithms. Similarly to changes in scale, the greater the change in perspective, the greater the drop in performance in terms of pass rate and homography estimation. However, especially between small and large view changes the drop is less severe than for scale changes. The larger drops for negative values of p can be attributed to the large image part not visible after transformation, remember Figure 4.27.

Figure 4.29 illustrates the effects of perspective change for combinations using the local feature detectors ORB and SURF. Again graphs for ORB15/BRIEF, ORB20k/RootSIFT and SURF/RootSIFT with DGF and K-VLD application and model estimation with ORSA 5 are displayed. Similar to the results for dense sampling, the larger the view change, the larger the drop in performance in terms of pass rate and homography estimation. Although the drop is less severe than for dense sampling, the lower initial performance of local feature detectors needs to be kept in mind. As a consequence, even though local feature detectors are more robust against view changes, individual performance values reached after perspective change is applied are comparable to those of dense sampling. Thus, if mostly view changes and only small scale changes are present in the underlying data, dense sampling is preferable. Instead, if perspective as well as scale changes are faced, a dense coverage of local feature detectors is favorable.

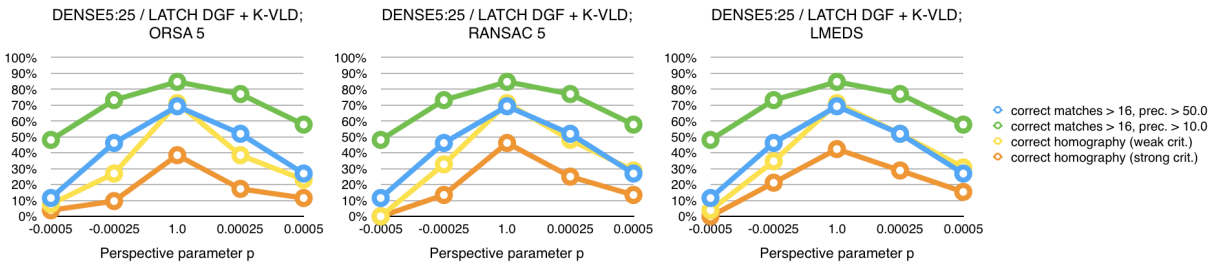


Figure 4.28: Effects of perspective change on pass rate and homography estimation for DENSE5:25/LATCH with DGF and K-VLD. For each perspective pass rate 16 with required precision above 10% and 50% as well as the percentage of good homographies according to the weak and strong criterion is displayed. All graphs illustrate, the larger the change in perspective, the larger the performance drop in pass rate and homography estimation, independent of the model estimation method used.

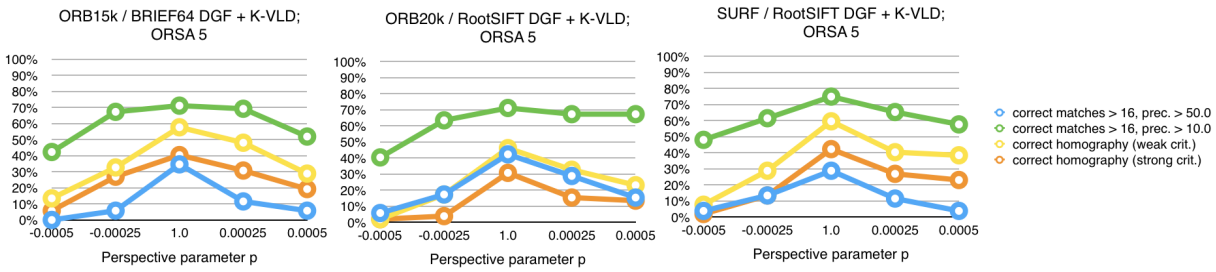


Figure 4.29: Effects of perspective change on pass rate and homography estimation for top combinations including the local feature detectors ORB and SURF. For each perspective pass rate 16 with required precision above 10% and 50% as well as the percentage of good homographies according to the weak and strong criterion is displayed. For local feature detectors, perspective change does result in small performance drops for pass rate and homography estimation.

4.3.7 Application Beyond Rephotography

Finally, we verify the applicability of the established pipeline to other challenging image pairs beyond rephotography and the nyc-grid dataset. To do so, we apply the top nine detector and descriptor combinations together with the proposed filtering mechanisms to the dataset of Hauagge and Snavely [2012]. This is composed of various challenging image pairs, including day and night images and images of different rendering styles.

The mean average precision values reached by Hauagge and Snavely [2012] are repeated in Table 4.5. Unfortunately, their mean average precision values are not directly comparable to the average precision reached after filtering, since they applied a variety of ratio thresholds for filtering and report mean average precision of all these tests. However, as reported previously, applying the ratio test to image pairs featuring major appearance changes leads to complete elimination of correct matches for many pairs, while other feature very high precision values. This may lead to good average precision values, but is undesirable if interested in correct alignment of as many images as possible.

Alternatively for comparison we combine one of the top performing detector and descriptor combination of Hauagge and Snavely [2012], namely GRID/SIFT, a dense sampling with keypoint spacing and scale of 25 pixels, with the proposed filters. The results of GRID/SIFT [Hauagge and Snavely, 2012] as well as all of the top nine combinations after applying DGF only and a combination of DGF and K-VLD are reported in Table 4.6. The left table shows results for the dataset of Hauagge and Snavely [2012], while the right table displays the corresponding results for the nyc-grid dataset. The pass rate depicted in the last column requires at least 16 correct matches and a precision value above 10% for each image pair.

	GRID	SIFT	SYM-I	SYM-G
Self-Sim.	0.29	0.14	0.12	0.16
SIFT	0.49	0.21	0.28	0.25
SYMD	0.41	0.22	0.20	0.25
SIFT-SYMD	0.58	0.28	0.35	0.36

Table 4.5: Mean average precision of different ratio thresholds as reported by Hauagge and Snavely [2012].

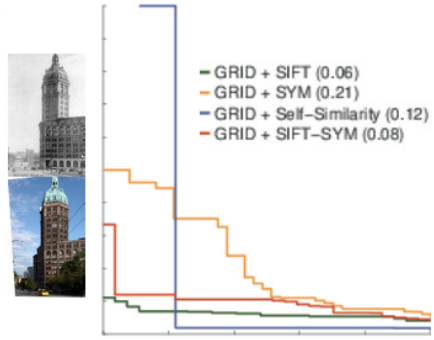
	none Prec.	DGF Prec.	DGF + K-VLD Prec.	DGF + K-VLD PassR.	none Prec.	DGF Prec.	DGF + K-VLD Prec.	DGF + K-VLD PassR.
GRID/SIFT [Hauagge,2012]	0.24	0.66	0.71	0.78	0.13	0.60	0.63	0.71
DENSE5:25/RootSIFT	0.24	0.69	0.84	0.96	0.11	0.53	0.66	0.91
DENSE5:25/LATCH64	0.13	0.71	0.85	0.96	0.06	0.61	0.72	0.84
ORB20k/RootSIFT	0.11	0.56	0.71	0.91	0.04	0.31	0.43	0.71
ORB20k/U-SURF	0.10	0.64	0.74	0.96	0.04	0.39	0.47	0.71
FAST/RootSIFT	0.09	0.56	0.63	0.87	0.03	0.30	0.34	0.71
SURF/RootSIFT	0.08	0.51	0.65	0.98	0.03	0.28	0.37	0.75
FAST/BRIEF64	0.06	0.53	0.61	0.87	0.02	0.24	0.32	0.69
ORB15k/BRIEF64	0.07	0.46	0.57	0.87	0.03	0.29	0.37	0.71
FAST/LATCH64	0.09	0.57	0.64	0.87	0.03	0.30	0.37	0.71

Table 4.6: Average precision results of the top detector and descriptor combinations on the dataset of Hauagge and Snavely [2012] (left) in comparison to results for the nyc-grid dataset (right). At the top we depict the average precision values the GRID/SIFT approach of Hauagge and Snavely [2012] would generate after applications of the filters proposed in this work. The reported pass rate (last columns) requires 16 correct matches and precision values above 10%.

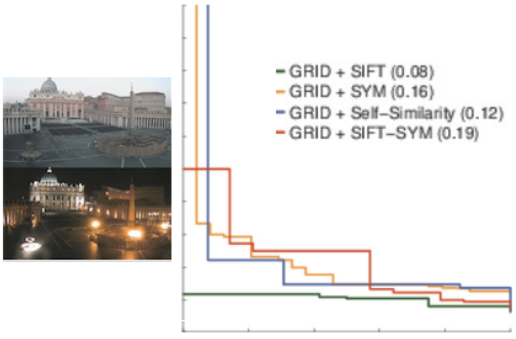
The left table shows that the top combinations, including dense sampling, outperform the GRID/SIFT approach of Hauagge and Snavely [2012]. While after application of DGF precision values vary about 5%, after additional K-VLD application, the performance difference is above 10%. Besides, the differences in pass rate are even more significant. In fact all the top combinations outperform GRID/SIFT [Hauagge and Snavely, 2012] in terms of pass rate, while the results for the four top combinations feature a pass rate up to 20% higher than that of GRID/SIFT.

Consequently, the presented approach is applicable to challenging image pairs beyond rephotography and outperforms previous approaches. Furthermore, comparing the left and right table it becomes clear, that for all methods the precision and pass rate values reached on the dataset of Hauagge and Snavely [2012] are higher than those achieved for the nyc-grid dataset. Hence, as expected, the nyc-grid dataset is more challenging due to the presence of major structural changes.

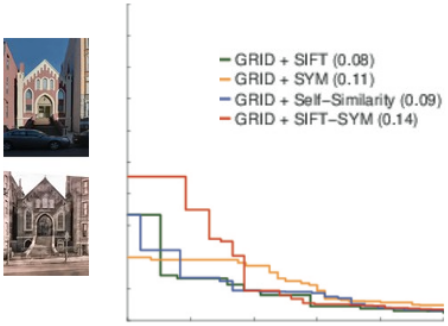
Finally, we compared the performance of our approaches on individual image pairs, which have shown to be extremely challenging based on the results of Hauagge and Snavely [2012]. A subset of these image pairs is displayed in Figure 4.30. The subset includes historic and modern, day and night images as well as drawings. All pairs feature rather low precision and recall curves in Hauagge and Snavely [2012], yet our pipeline is able to generate high precision values after match filter application. Indeed, the precision reached by dense sampling with RootSIFT descriptor and both DGF and K-VLD application, is 100% for 3 out of 4 of the examples. This confirms, that the presented pipeline is applicable to image pairs beyond rephotography, and outperforms previous approaches on challenging image pairs.



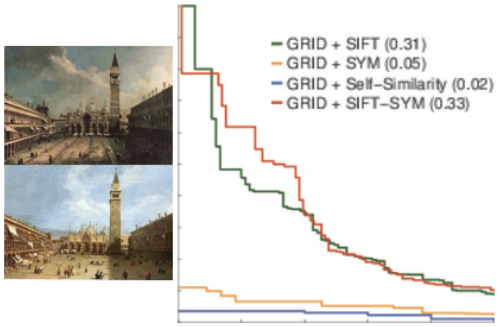
	DGF	K-VLD	both
DENSE5:25/RootSIFT	0.97	0.83	1.00
ORB20k/U-SURF	0.60	0.62	0.76
SURF/RootSIFT	0.54	0.49	0.62
FAST/LATCH64	0.59	0.18	0.59



	DGF	K-VLD	both
DENSE5:25/RootSIFT	0.86	0.92	0.95
ORB20k/U-SURF	0.10	0.00	0.28
SURF/RootSIFT	0.30	0.39	0.59
FAST/LATCH64	0.88	0.56	1.00



	DGF	K-VLD	both
DENSE5:25/RootSIFT	0.91	0.77	1.00
ORB20k/U-SURF	0.08	0.00	0.20
SURF/RootSIFT	0.45	0.37	0.59
FAST/LATCH64	0.48	0.34	0.60



	DGF	K-VLD	both
DENSE5:25/RootSIFT	1.00	0.61	1.00
ORB20k/U-SURF	0.30	0.31	0.36
SURF/RootSIFT	0.13	0.14	0.17
FAST/LATCH64	0.02	0.30	0.02

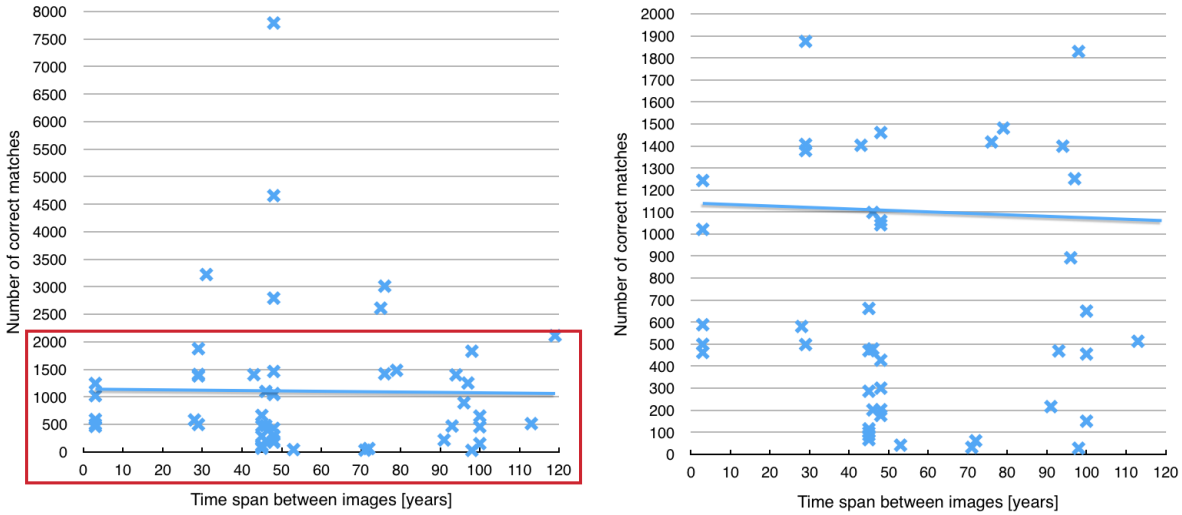
Figure 4.30: Comparison of results for individual image pairs from Hauagge and Snavely [2012]. At the top the image pairs and their respective recall and precision curves for dense sampling as reported by Hauagge and Snavely [2012] are displayed. At the bottom the precision values of a variety of detector and descriptor combinations reached after match filtering are shown. For the majority of image pairs the proposed pipeline reaches high precision values.

4.4 Identification of Critical Image Pairs

In the previous sections we were able to significantly optimize the performance of classic feature combinations by varying detector and descriptor parameters and applying appropriate filters. Nonetheless, the evaluation showed that for almost one third of image pairs achieving good alignment is impossible or at least very challenging with the proposed methods, remember the homography estimation results of Figure 4.22. In this section we identify these images and the main difficulties they pose to image matching, to gain ideas for the focus of future research.

Correct Matches vs. Time Span

First of all we check the statement of Schindler and Dellaert [2012] that there is an inverse relationship between the number of correct matches and the time span between images. Figure 4.31 illustrates this relationship DENSE5:25/LATCH64 before any filter application. On the left



(a) Full Diagram **(b)** Enlargement

Figure 4.31: Illustration of the correlation between the number of correct matches and the time span of an image pair for DENSE5:25/LATCH. (a) depicts the full graph, while (b) shows only the bottom part containing most of the image pairs. Both illustrate that a long time span between an image pair does not necessarily have a negative influence on the number of correct matches. Instead many images with a time span of approximately 100 years feature many correct matches, while a great portion of images taken less than 50 years apart has only very few correct correspondences.

the full diagram is shown while on the right only the bottom part containing most image pairs is visible. Both graphs shows that no significant linear correlation between number of correct matches and time span exists in our dataset. This is also shown by the trendline. Instead, diverse images of rather short time spans (ca. 50 years) show a low number of correct feature correspondences, while many images with a time span of approximately 100 years feature many correct matches. Thus, we can not verify the statement of Schindler and Dellaert [2012] for our top feature detector and descriptor combinations. Note, that Schindler and Dellaert [2012] based their statement on applying the SIFT detector and descriptor with a strict ratio threshold of 0.6.

Set of Critical Image Pairs

To analyze the main difficulties faced by aligning historic to modern images, at first we identify image pairs for which good homography estimation is extremely challenging. In detail, for each of the top nine detector and descriptor combinations, we choose the best performing combination of match filtering and model estimation, remember Figure 4.22. Thus, we have 9 different full pipelines, including feature detection, description, match filtering and model estimation. Now for each image pair we determine whether these pipelines output a good homography (weak criterion) and sort images pairs by the total number of good homographies.

Figure 4.32 illustrates the distribution of good homography estimation across the dataset. It shows that 20 image pairs (almost 40%) are correctly matched by each pipeline and an additional 17 image pairs (33%) are well aligned for the majority of pipelines (5 to 8). Thus, for 70% of the dataset model estimation is generally successful, while for 9 image pairs (17%) model estimation fails with each pipeline. These are the most challenging image pairs. For another 6 pairs (12%) a good homography estimation is only achieved with individual pipelines, so we attribute these

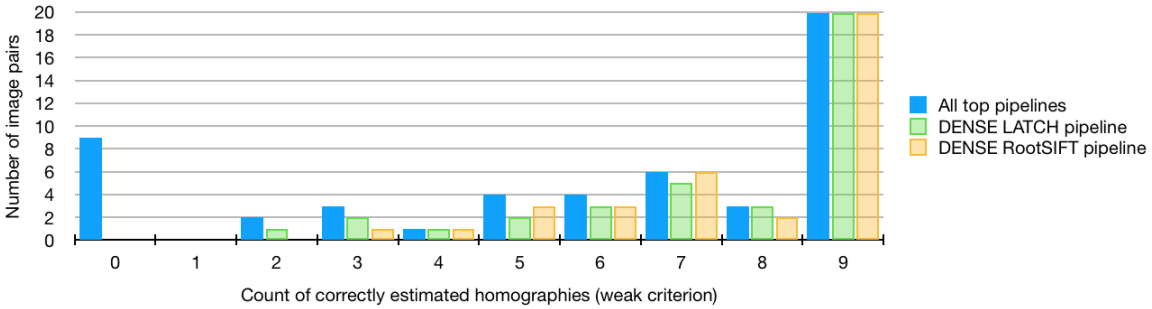


Figure 4.32: Illustration of the distribution of good homography estimation across the nyc-grid dataset. The blue bar depicts how many image pairs show 0 to 9 good homography estimations if all top nine pipelines are considered. Instead, the green and yellow bar depict the number of images pairs from each subset, represented by the blue bar, for which homography estimation succeeds with DENSE5:25/LATCH (DGF + K-VLD + LMEDS) and DENSE5:25/RootSIFT (DGF + K-VLD + ORSA).

Figure 4.33: Examples of critical image pairs where the challenges to image matching are obvious. The left image pair suffers from high occlusion due to the large tree in front of the building. For the top right pair, only the background of both images stayed the same, while in the foreground the road and fence changed their pathway and the trees in the center developed. Even more difficult is the scenery in the bottom right image pair. Here the only stable element seems to be the street light and the wall it is placed on.

Figure 4.34: More examples of critical image pairs. At first glance the historic and modern scenery looks very similar, yet taking a closer look one realizes how much the presented buildings changed. In the left pair, almost all building on the left and right side of the street changed their entire facade or were newly constructed. In the right images, only the old tower like building, apart from its spire, and the one right next to it have been preserved across the years, while all others changed their facade, where enlarged or replaced.

Figure 4.35: Examples of image pairs where none or very few accidental correct matches are identified.

to the set of critical image pairs as well. As a result, a total of approximately 30% of the image pairs of the dataset are very difficult to align and belong to the set of critical image pairs.

Figure 4.33 to 4.38 show image pairs belonging to this set. For some image pairs it is obvious, where the challenges in matching arise from. Figure 4.33 shows some examples of image pairs, suffering from high occlusion due to vegetation and structural changes to a scene. In other image pairs the great structural changes and changed facades of buildings only become visible if taking a closer look, see also Figure 4.34.

Taking a closer look at the homography estimation results of all critical image pairs, three main reasons for the failure of good model estimation can be identified. The first is an extremely low total number of correct matches. Figure 4.35 shows two image pairs with the remaining correct matches, after applying DENSE5:25/LATCH with DGF followed by K-VLD. In the first image no correct match remains after filtering, while in the second image four accidental correct matches are visible clustered at a single region. Thus in case of the first pair the model estimation algorithm has no chance to pick the correct model, while for the second image pair, with only 4 correct matches and a precision of 6.8%, correct model estimation is highly unlikely.

The second reason for model estimation failure is the clustering of correct matches to a single or few small regions. Two examples are presented in Figure 4.36 and Figure 4.37. The image pair in Figure 4.36 contains 144 correct matches and features a precision of 82%, which seems perfect for image alignment. Indeed, model estimation uses all correct matches, but they are clustered to a single region, the left part of the statue. Consequently, this is well aligned by the estimated homography, while the surroundings suffer from bad alignment. This is nicely visible at the marked regions (red circles) in the image overlay, consisting of the old image superimposed by the edges of the transformed new image. The image in Figure 4.37 contains 19 correct matches and features a precision of 56%. This outperforms the thresholds required for pass rate as well, yet again all correct matches are clustered at two small regions in one part of the image, which results in bad alignment for the rest of the image.

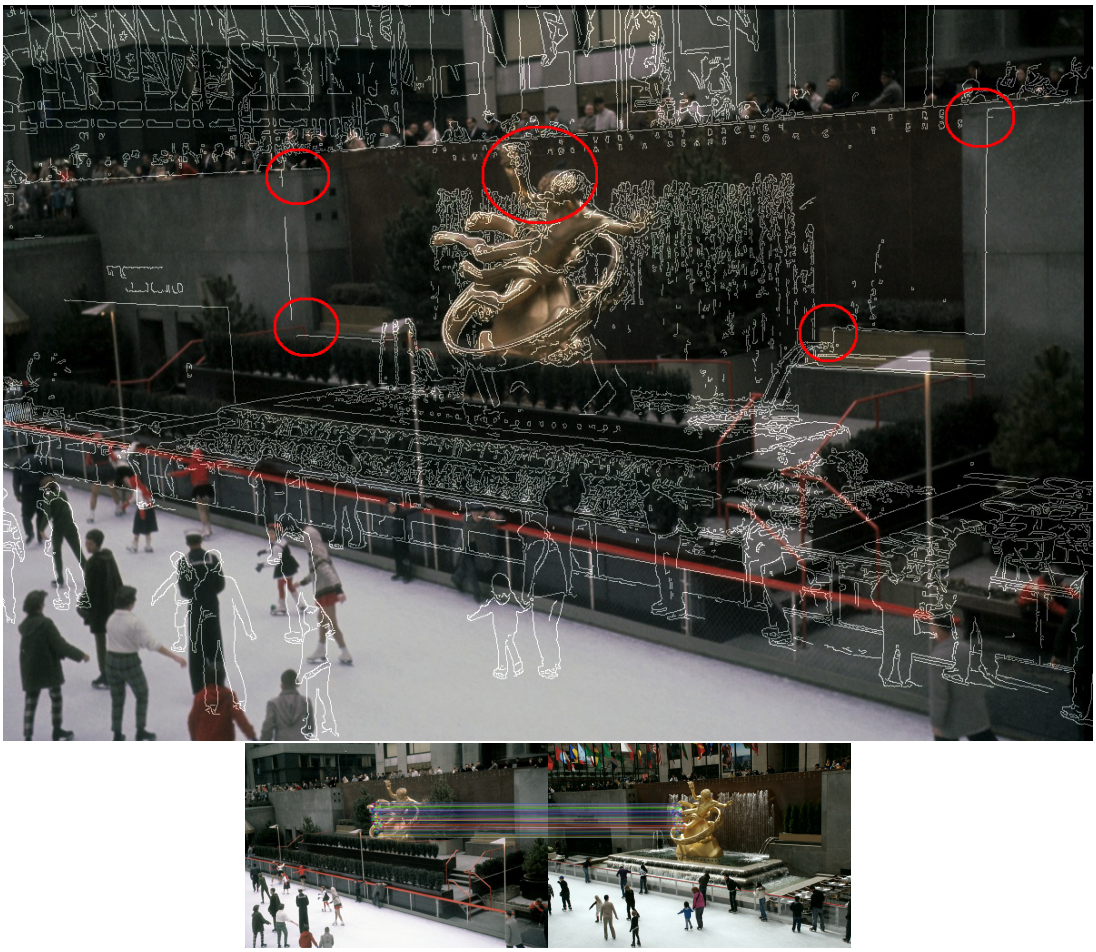


Figure 4.36: Example of an image pair, where correct matches are clustered at a small region. Here the left part of the statue. This is correctly aligned, while surroundings suffer from misalignment. Red circles mark regions of bad alignment in the overlay.

Third, model estimation fails if all correct matches lie in one part of the image, for instance the left half of the image. Figure 4.38 presents an example. For the presented pair the number of correct matches is 268 with a precision of 91%. Yet, all correct matches belong to the row of houses on the left side of the street. Consequently, this is perfectly aligned, while the building on the right side is tilted.

The first reason for failure, to few correct matches, is already indicated by pass rate requiring at least 16 correct matches. Instead, the image pairs, where model estimation fails, due to the distribution of correct matches, commonly pass both pass rate thresholds, the minimum number of 16 correct matches as well as the required precision of 10%. Thus, pass rate is to optimistic and not able to adequately predict the percentage of images for which a good model estimation will be achieved. This observation is in line with the low results achieved during model estimation in Section 4.3.5.

Figure 4.37: Second example of an image pair, where correct matches are clustered at a small region. This is correctly aligned, while surroundings are ignored. Red circles mark regions of bad alignment.

4.5 Summary

With regard to the main questions asked in the beginning the results of the conducted evaluation can be summarized as follows.

General Suitability of Classic Feature Detectors and Descriptors

First of all, keypoints detected by local feature detectors are not stable across the long time spans faced in historic to modern image matching. Thus, as proposed by Fernando et al. [2015] and Stylianou et al. [2015] a dense sampling of keypoints is preferable. Recently further works have demonstrated the advantages of dense sampling in the context of matching images with challenging viewing conditions. Among these are day and night images [Sattler et al., 2018; Zhou et al., 2016] as well as weakly textured indoor scenes with many repetitive elements [Taira et al., 2018].

However, densely sampled features are unsuitable in the presence of large scale changes. This is not relevant in the context of historic rephotography, while for other areas of application it can be. In this case we propose a dense coverage by local feature detectors such as ORB and SURF, which produce keypoints at different scales. This can be achieved by decreasing selection thresholds resulting in a detection of as many keypoints as during dense sampling.

Figure 4.38: Example of an image pair, where correct matches are limited to a larger region in the left half of the image. These region is matched well, but the periphery (red circles) and right half of the image are distorted (tilt of the historic building on the right).

Concerning feature description in the context of historic to modern imagery several classic descriptors showed good performance. These include RootSIFT and U-SURF as floating point descriptors, but also the binary descriptors LATCH and BRIEF. In the case of binary descriptors we recommend extending descriptor length to 64 elements to enhance descriptive power. With this LATCH showed an overall performance as good as that of RootSIFT at a fraction of its computational complexity.

Nonetheless, independent of the specific detector and descriptor combination used, historic to modern image pairs, suffer from very low precision values after correspondence computation. This is a result of the large appearance and structural changes present in these image pairs. These impede the identification of correct local correspondences, while additionally inducing accidental false correspondences. Thus, good filters are required to reduce the number of outliers.

Filtering Approaches for Compensating Low Matching Performance

Common correspondence filters, such as the ratio test, based on descriptor distance fail in the context of historic to modern image matching. This is because even correct correspondences feature large descriptor distances due to the great appearance changes of these images. Thus, comparing descriptor distances of the best and second best match is no longer meaningful for outlier detection. Similar effects can be observed if the density of the descriptor space is massively

increased, as in the context of visual localization in large 3D models [Li et al., 2012]. In this case descriptor distances of correct correspondences are still small, but due to the large descriptor space there are also a lot of accidental correspondences. This has a comparable negative effect on the ratio test.

Instead, filters based on geometry and further additional constraints are able to boost matching performance in terms of precision. Especially DGF [Ali and Whitehead, 2014; Roth and Whitehead, 2000], purely based on geometric constraints, performed well even facing extremely low inlier ratios of 1 to 3%. Another successful filter is K-VLD [Liu and Marlet, 2012], based on geometric as well as photometric constraints. Yet, it eliminates more correct matches than DGF. Further geometric filters have not been part of this evaluation, yet for such we expect comparable results.

Alignment via Model Estimation

The final goal of rephotography is to correctly align a historic image to the modern scene. This evaluation achieves this by estimating a homographic transformation between both images. To evaluate these homographies a new measure is proposed. This is based on comparing the surface and the corners of the transformed image to the results of the ground truth transformation. With these measures we establish a strong and a weak criterion allowing to distinguish between good and bad homographic alignment.

This measure allows to assess the results of different model estimation algorithms, including ORSA [Moisan and Stival, 2004], RANSAC [Fischler and Bolles, 1981] and LMedS[Rousseeuw, 1984]. These show that there is no model estimation approach which significantly outperforms the others. In detail ORSA often tends to generate models overfitted to very few correspondences, while RANSAC and LMedS require high precision values (mostly $> 50\%$) for correct model estimation. Furthermore, since outlier filtering by model estimation is rather unreliable, we conclude that further outlier filtering before model estimation is very important.

Major Difficulties For Matching

Overall, successful alignment of historic and modern images is achieved for the majority of images of the nyc-grid dataset. Yet, for about 30% of image pairs matching is extremely difficult. The future goal is to be able to match these challenging image pairs.

The major cause for failures in image alignment is a low number of correct matches. In part this results from the great occlusions, due to structural changes and vegetation, present in historic and modern image pairs. However, observation reveals, that more repeated structures are visible than those detected and related via correspondences. Thus, a lack of adequate feature description and matching also causes failures.

Another issue is the clustering of correct matches to single or very few regions. With regard to the global image transformation, each of these clusters is only as informative as a single correct match. Thus, despite an apparent high number of correct correspondences, final image alignment fails. Again, this is a direct result of high occlusion and the failure to adequately match keypoints under strong appearance changes.

This evaluation uses pass rate to measure the prospect of successful alignment. Pass rate is able to detect a low number of correct matches. This is its advantage to measuring precision only. However, it is unable to predict alignment failure due to clustering of correct matches. As a result, it is to optimistic in predicting the success of model estimation. This explains the low values of good homography in comparison to pass rate for all top combinations.

Further Remarks

Furthermore, we showed that matching historic to modern images is significantly more difficult if facing large scale or viewpoint changes. Consequently, we agree with Torii et al. [2015] that matching is easier for slight shift in viewpoint. However, since in rephotography small changes in scale and viewpoint are the norm, at the moment this is not a challenge for this thesis.

On the other hand, we can not confirm the statement of Schindler and Dellaert [2012] and Wolfe [2013], that image registration becomes more difficult as the time span between images increases. Instead, the difficulty of matching highly depends on the amount of structural change in an image pair. Often long time spans coincide with more structural change. However, some scenes also experience major changes due to construction or new vegetation across only a few years. As a consequence matching these is difficult as well.

4.6 Outlook

To enhance alignment, especially for challenging historic and modern image pairs, the following should be the focus of future work in this area.

At first, the generation of too few correct correspondences, resulting in a low number of correct matches and/or their clustering, needs to be addressed. This might be achieved by new feature descriptors which are more robust to severe appearance changes. Another alternative is to reduce the number of possible matches for each keypoint via prefiltering and/or preselection. For instance keypoints on mobile objects could be filtered prior to performing nearest neighborhood matching to reduce the number of false correspondences.

Second, advanced correspondence filtering methods for outlier reduction are required. These need to be able to handle the very low precision values present in historic to modern image matching. Furthermore, their rate of correct correspondence rejection needs to be comparatively low to avoid the rejection of single correct matches in areas largely occupied by false matches. Only if singular correct correspondences are preserved, successful alignment of the entire image, instead of only few regions, can be achieved.

Third, to evaluate the success of detection and preservation of correct correspondences across the entire image, new measuring approaches are required. These need to predict the success of homography estimation, not only based on precision and the number of correct matches, but also their location and distribution. In case such a measure is established, a model estimation via RANSAC, favoring solutions including correspondences spread across the entire image, might be able to enhance homography estimation.

Chapter 5

Measuring Match Distribution

The measure of *pass rate*, developed by Gat et al. [2011], tries to predict the success of model estimation for an image pair. To do so it defines thresholds for the minimum number of correct correspondences and the minimum precision. If both thresholds are exceeded successful model estimation is assumed.

The previous evaluation, presented in Chapter 4, showed that pass rate is too optimistic in predicting the success of homography estimation in the challenging context of historic to modern image matching. This is, since it ignores the issues of clustered correspondences and ill coverage of the entire image.

In this chapter we develop an alternative *advanced pass rate*, that does not only measure the number of correct correspondences but also their distribution. Thus, it is able to identify images for which model estimation is likely to fail due to correspondence clustering. Finally, we show that advanced pass rate is more reliable in predicting the outcomes of model estimation.

5.1 Literature

To the author's knowledge, in the literature only few works exist, which address the need to measure correspondence distribution in addition to simple correspondence count. Irschara et al. [2009] measure spatial distribution of features across an image to enhance location recognition in 3D point clouds. They note that for successful image registration uniformly distributed image features are superior to feature clusters, see also Figure 5.1. Their approach weighs the number of correspondences by the covered image fraction resulting in an effective correspondence count.

Schönberger and Frahm [2016] use a spatial distribution score to choose the next best view for structure-from-motion reconstruction. They share the idea of Irschara et al. [2009], that a more uniform distribution of features in an image enhances reconstruction. For measuring distribution they split the image into a grid with K_l cells in each dimension. If one or more points are visible in a cell this is regarded as *full*, otherwise it is *empty*. As a result, point clusters are only counted once or twice depending on their dimensions. To better reflect the distribution of occupied cells, their final score accumulates results of a multi-resolution pyramid of grids. In detail the image is split at L different resolutions $l = 1...L$ with $K_l = 2^l$. At each level an occupied cell contributes to the final score with a weight $w_l = K_l$. Figure 5.2 shows some examples of different point distributions and their score.

Figure 5.2 nicely illustrates that if more cells are occupied in the highest resolution layer K_L the overall score is higher, compare left to right. Furthermore, it seems if the number of occupied cells in layer K_L remains the same, but these cells are more evenly distributed across the image,

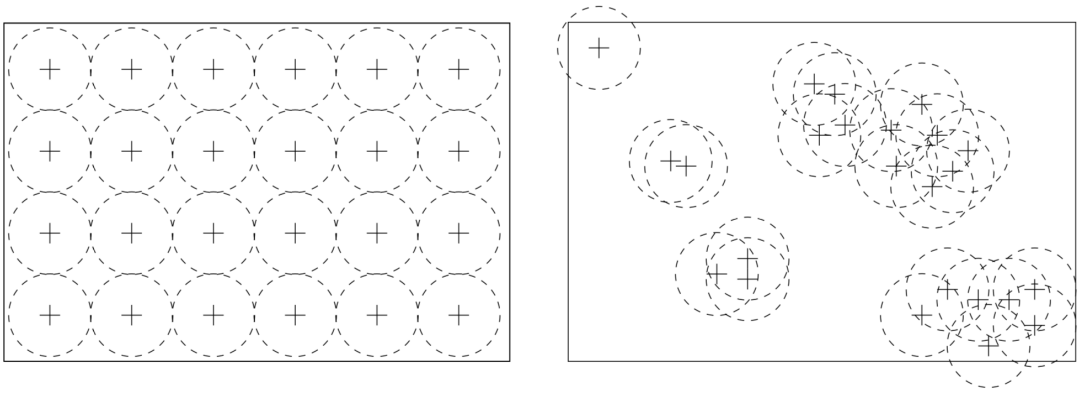


Figure 5.1: Illustration of the coverage of uniformly distributed (left) vs. non-uniformly distributed image features (right). For image registration a uniform distribution of image features is preferable to feature clusters. Image from Irschara et al. [2009]

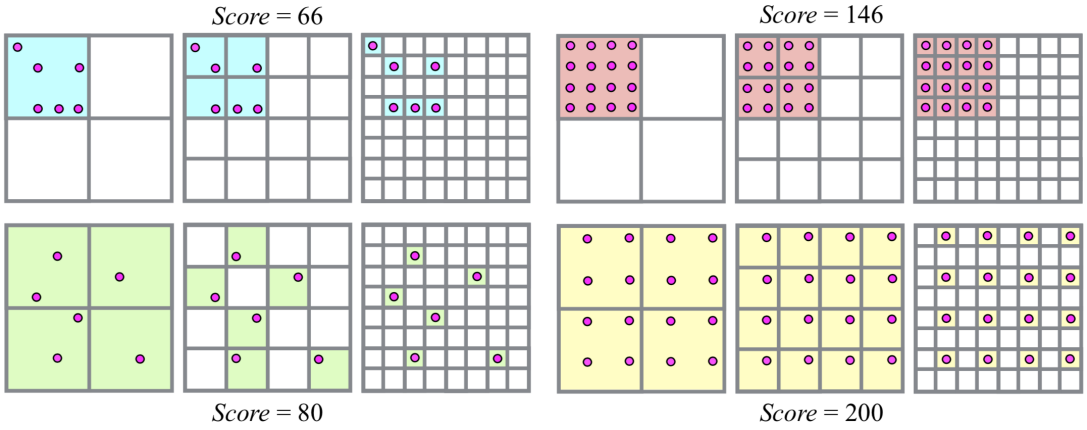


Figure 5.2: Illustration of different point distributions and their corresponding scores. Image from Schönberger and Frahm [2016].

the score is higher as well, compare top to bottom. However, Figure 5.3 shows two examples with only four points, whose distributions are very different, yet their score according to Schönberger and Frahm [2016] is the same. While the points in the left image are well distributed and able to represent a global image transformation, the points in the right image form a single cluster at the image center. Thus, the left image should have a significantly higher score than the right image. Another example of misleading scores is already illustrated in Figure 5.2, if comparing the point distributions at the bottom left and top right. The score of the top right example (red) is significantly higher, this is mainly due to the larger number of occupied cells at the highest resolution. Yet, the points in this example only cover one fourth of the entire image, while the points in the bottom left example (green) are distributed across the entire image and only miss its periphery. Consequently, if the goal is global image registration, the distribution on the bottom left seems preferable, despite their low score.

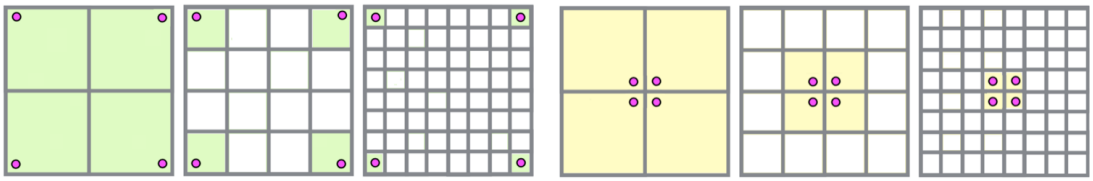


Figure 5.3: Illustration of two different point distributions with the same score, calculated by the method of Schönberger and Frahm [2016]. Despite their low score, the distribution in the left image is more valuable for reconstruction and other model estimation algorithms aiming at computing a global image transformation. This is, because the points on the left are distributed across the image, while the points in the right simply form a large cluster.

5.2 Our Method

In the previous section we described previous approaches to measure point distribution across an image and illustrated their weaknesses. Now we present our method and show that it is able to adequately distinguish between the favorable and non-favorable examples presented previously. In summary, a large span width of points should lead to a higher score, while more points with the same span width are preferable. However, if many points cover only a small part of the entire image, a solution with less points spanning a larger area of the image should result in a higher score.

Effective Matches and Effective Coverage

As a first step, we follow the approach of Schönberger and Frahm [2016], impose a grid on each image and count only a single point per cell. In our case each of these points belongs to a correct correspondence, thus we call them *effective matches*. Depending on its format, we impose a grid of 12x8 or 8x12 cells on each query image resulting in a total of 108 image regions, independent of the original image resolution. This is necessary to allow the comparison of scores between images of varying resolution. Furthermore, opposed to Schönberger and Frahm [2016], we decided to use a rectangular grid, since almost all photographs we deal with feature a rectangular format of approximately 4:3, which is well reflected by our grid size.

Now, for each cell only one correct correspondence is counted as effective. Thus, single correct matches in four different image regions are ranked as valuable as four inlier clusters, see also Figure 5.4. Finally *effective coverage* depicts the percentage of cells that contain an effective match. Hence, if an image features an effective correspondence coverage of 20%, 20% of the cells of the imposed grid contain at least one correct correspondence.

As mentioned previously, by counting effective matches local image clusters are dealt with, but the simple measure of coverage is not able to depict different distributions of effective matches. Consequently, this needs to be accessed separately.

Effective Match Distribution

To measure effective match distribution our first idea was to average between the distances of all points in the query image, belonging to an effective match. However, while this measure is able to compare sparse point distributions with each other very well, average distance decreases with every additional point placed between others. For an illustration please refer to Figure 5.5. This is undesirable, since additional effective matches are always valuable for model estimation.

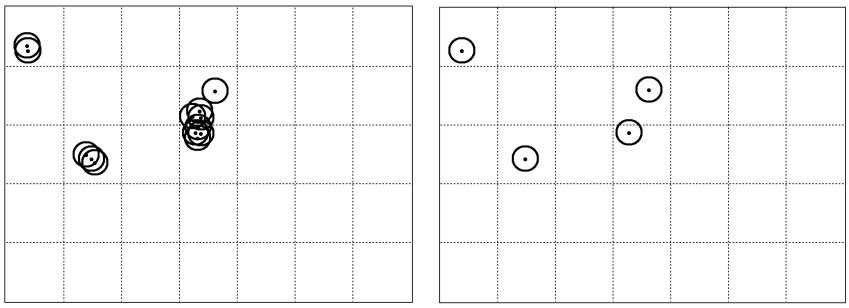


Figure 5.4: Illustration of our measure of effective match count. While the left image contains many correct matches these are clustered at few regions. The right image instead contains only four more evenly distributed matches. For homography estimation only distributed correct matches are valuable. Thus, we consider only a single match per region as effective, resulting in a total effective match count of four for both images.

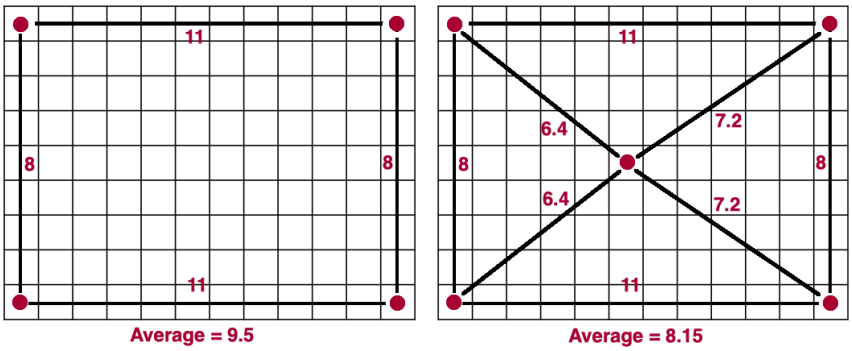


Figure 5.5: Results of calculating average distance between effective matches for two different point distributions.

Therefore, in the examples shown in Figure 5.5 the left distribution should receive a lower score than the one on the right. Consequently, we developed a more advanced approach, which is described in the following.

At first, we generate a second grid with the same dimensions as the grid marking effective matches. However, this grid does not contain the values *empty* and *full*, but integer values. At the beginning all cells are initialized with the value $m = max(gridwidth, gridheight) - 1$, in our case $m = 11$. Furthermore, the $Score$ is initialized with 0.

Now, among all cells containing an effective match, the one with the highest value is selected. In case more than one cell contains an equally high value, the one most close to the periphery is selected. To determine the closeness to the periphery, the number of rows and columns towards the edges are counted. This means each corner has a distance of 0, while its direct neighbors have a distance of 1 and its diagonal neighbor a distance of 2. If two or more of the relevant cells, with maximum value also have the same minimum distance to the periphery, the first cell found when advancing from left to right and top to bottom is selected.

The value of the selected cell is added to the $Score$, and the values in its surroundings are updated according to the following scheme. The direct and diagonal neighbors of the selected cell receive the value 1, forming a ring of 1's around the cell. The next ring receives the value 2, the third ring the value 3 and so on, until the edge or a cell already containing a smaller value is reached. Consequently, cell values can only decrease.

11	11	11	11	11	11	11	11	11	11	11	11
11	11	11	11	11	11	11	11	11	11	11	11
11	11	11	11	11	11	11	11	11	11	11	11
11	11	11	11	11	11	11	11	11	11	11	11
11	11	11	11	11	11	11	11	11	11	11	11
11	11	11	11	11	11	11	11	11	11	11	11
11	11	11	11	11	11	11	11	11	11	11	11
11	11	11	11	11	11	11	11	11	11	11	11
11	11	11	11	11	11	11	11	11	11	11	11

(a) Initialization: In the beginning all grid cells are initialized with the same value and the score is initialized with 0. For visualization grid cells featuring an effective match are highlighted in green. Now among all cells containing an effective match the one with the highest value, located most far to the periphery is selected.

11	1	2	3	4	5	6	7	8	9	10	11
1	1	2	3	4	5	6	7	8	9	10	11
2	2	2	3	4	5	6	7	8	9	10	11
3	3	3	3	4	5	6	7	8	9	10	11
4	4	4	4	4	5	6	7	8	9	10	11
5	5	5	5	5	5	6	7	8	9	10	11
6	6	6	6	6	6	6	7	8	9	10	11
7	7	7	7	7	7	7	7	8	9	10	11
8	8	8	8	8	8	8	8	8	9	10	11

(b) Iteration 1: The value of the selected cell (top left corner, highlighted in dark green) is added to the score, and the values in its surroundings are changed according to the following scheme. Direct and diagonal neighbors receive the value 1, forming a ring of ones around the selected cell. The next ring receives the value 2 and so on, till the edge or a cell having a smaller value is reached.

11	1	2	3	4	5	5	4	3	2	1	11
1	1	2	3	4	5	5	4	3	2	1	1
2	2	2	3	4	5	5	4	3	2	2	2
3	3	3	3	4	5	5	4	3	3	3	3
4	4	4	4	4	5	5	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8	8	8	8

(c) Iteration 2: Again the effective match with the highest value, closest to the periphery, is selected and its surrounding values are decreased.

11	1	2	3	4	5	5	4	3	2	1	11
1	1	2	3	4	5	5	4	3	2	1	1
2	2	2	3	4	5	5	4	3	2	2	2
3	3	3	3	4	5	5	4	3	3	3	3
4	4	4	4	4	5	5	4	3	3	3	3
3	3	3	3	4	5	5	4	3	2	2	2
2	2	2	3	4	5	5	4	3	2	1	1
1	1	2	3	4	5	5	4	3	2	1	7
8	1	2	3	4	5	5	4	3	2	1	1

(d) Iteration 4: At this stage all effective matches at the corners have been selected and contributed to the $Score = 37$ with their respective values.

11	1	2	3	3	3	3	3	3	2	1	11
1	1	2	3	2	2	2	2	2	2	1	1
2	2	2	3	2	1	1	1	2	2	2	2
3	3	3	3	2	1	5	1	2	3	3	3
4	4	4	3	2	1	1	1	2	3	3	3
3	3	3	3	2	2	2	2	2	2	2	2
2	2	2	3	3	3	3	3	3	2	1	1
1	1	2	3	4	4	4	4	3	2	1	7
8	1	2	3	4	5	5	4	3	2	1	1

(e) Iteration 5: Now the effective match with the highest value from the interior is selected and its surrounding values are updated as long as they do not contain lower values already.

11	1	2	3	3	2	2	2	2	2	1	11
1	1	2	3	2	2	1	1	1	2	1	1
2	2	2	2	2	1	1	1	1	2	2	2
3	2	1	1	1	1	5	1	1	2	3	3
3	2	1	2	1	1	1	1	2	2	3	3
3	2	1	1	1	1	1	1	1	2	2	2
2	2	2	1	3	1	1	3	1	2	1	1
1	1	2	1	1	1	1	1	1	2	1	7
8	1	2	2	2	2	2	2	2	2	1	1

(f) Termination: The algorithm terminates when all effective matches have been selected and added to the score with their respective values. In this example the final $Score = 51$.

Figure 5.6: Illustration of our algorithm generating a score for effective match distribution.

Now, again the effective match with the highest value closest to the periphery is selected, added to the score and its surrounding values are decreased. This procedure is repeated until all effective matches have been selected and contributed to the final *Score* with their respective value. For an illustration of our algorithm please refer to Figure 5.6. The final *Score* ranks the distribution of all effective matches, while higher scores indicate a better distribution for global transformation estimation.

Comparison to Schönberger and Frahm [2016]

Figure 5.7 compares the scores of our algorithm with those of Schönberger and Frahm [2016], for all distributions previously presented in Figure 5.2 and 5.3. It shows that our method clearly identifies the best (central right) but also the worst distribution (bottom right). And, opposed to the algorithm of Schönberger and Frahm [2016], it distinguishes between the two distributions at the bottom and favors the left one, as desired. Furthermore, if distributions are similar, configurations with more points are preferred, as illustrated by the top two examples.

Room for improvement only remains, when comparing the score of the top right (red), with that of the central left and bottom left distribution (both in green). For global model estimation the bottom left distribution which features points in all image corners is preferable to the top right distribution only covering one fourth of the image. However, due to the increased number of points in the top combination, both have the same final score. Yet, in practice it is common that a distribution as the one on the bottom left additionally features one or more effective matches in the center. In this case its score increases and it outperforms the score of the top right distribution.

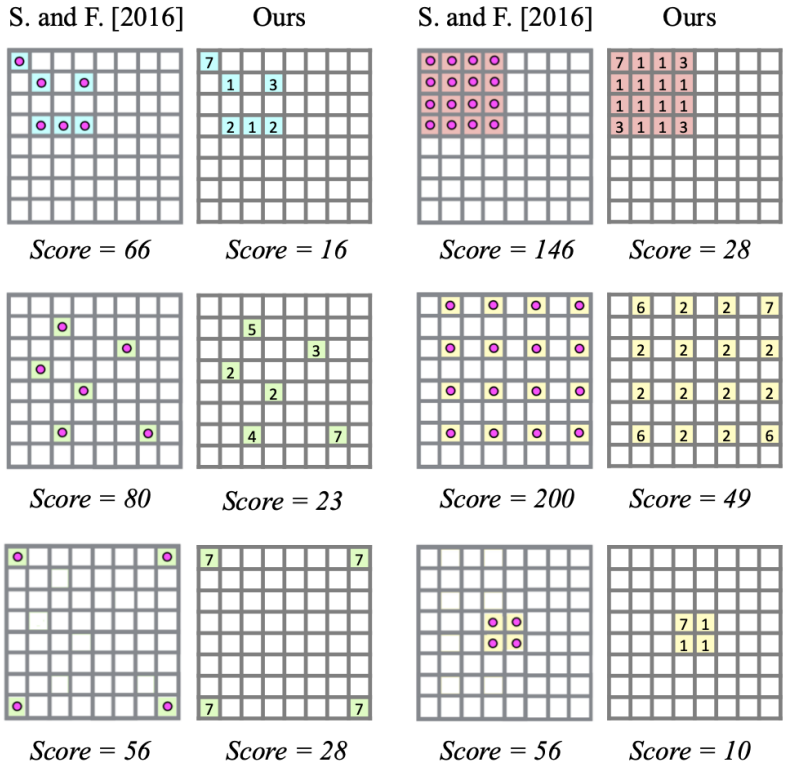


Figure 5.7: Results of our method compared to those of Schönberger and Frahm [2016].

More of an issue is the comparison between the top right and the central left distribution. While the central left one covers a greater fraction of the entire image it only receives a score of 23 opposed to a score of 28 for covering one fourth of the image. However, the difference in ratio between these two distribution is only 0.82 compared to 0.55 for the method of Schönberger and Frahm [2016]. Thus, also for comparing these difficult distributions our method is an improvement and if a few more points are added to the central left distribution, this receives a higher score as well.

Advanced Pass Rate

After generating a score for effective match distribution, we want to establish an advanced pass rate based on this score. To do so, we determine new thresholds for precision and effective match distribution. If both thresholds are exceeded successful model estimation be expected.

For establishing these thresholds, we matched the image pairs of the nyc-grid dataset with three different features, including DENSE/RootSIFT, DENSE/LATCH and Relocalized Dense CNN, which all show good performances for our dataset. A detailed explanation of Relocalized Dense CNN follows in Chapter 6. For homography estimation LMedS and RANSAC were used with each feature, resulting in a total of 312 examples. For these we compare between good and bad homography estimation via the weak criterion, remember Section 4.3.5. Results are shown in Figure 5.8, which plots good and bad homography estimation results against precision and effective match distribution score. The gray area marks all examples who exceed the thresholds we fixed for advanced pass rate, which requires a precision of 15% and a minimum effective match distribution score of 35.

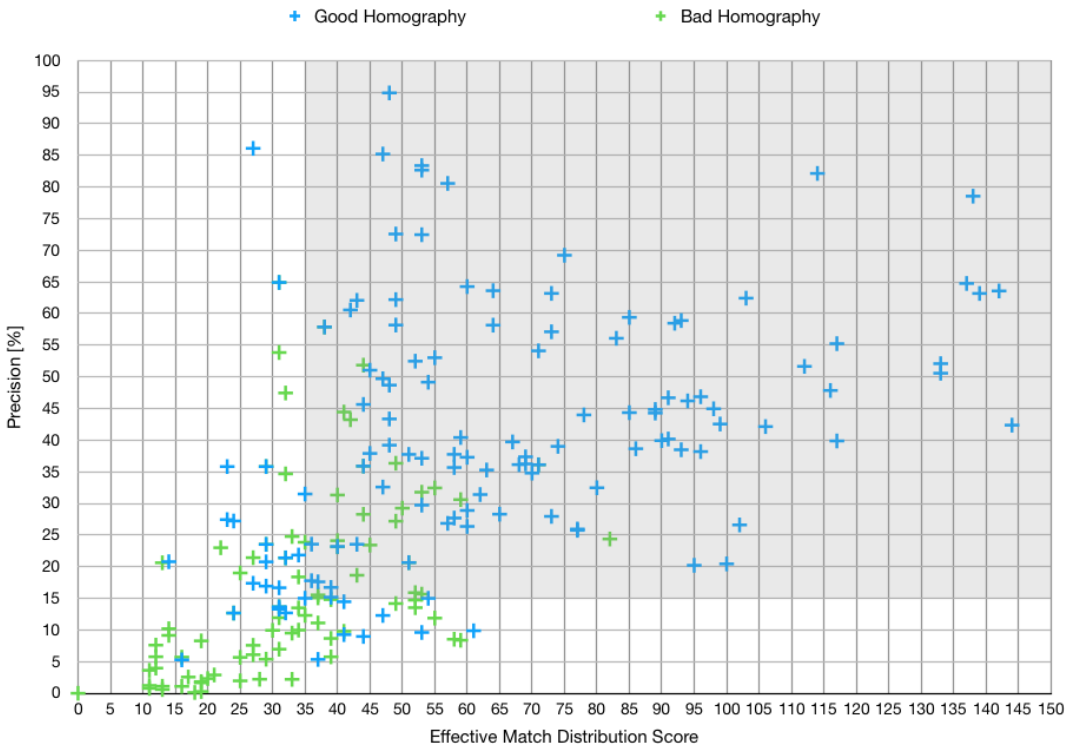


Figure 5.8: Illustration of good and bad homography estimation in terms of precision and effective match distribution score. The gray area marks all examples who exceed the thresholds of advanced pass rate.

Feature	Model Estimation Method	Good Hom. Rate	Advanced Pass Rate	Pass Rate effM>=8 prec>15.0	original Pass Rate corrM>16 prec>10.0
Reloc.Dense CNN	LMEDS	69.2	69.2	78.8	90.4
Reloc.Dense CNN	RANSAC	71.2	69.2	78.8	90.4
Dense LATCH	LMEDS	61.5	59.6	78.8	84.6
Dense LATCH	RANSAC	67.3	63.5	78.8	84.6
Dense RootSIFT	LMEDS	48.1	53.8	55.8	67.3
Dense RootSIFT	RANSAC	50.0	51.9	55.8	65.4
Avg. Diff. to Good Hom. Rate			2.6	9.9	19.2
Standard Deviation			2.0	4.1	2.7

Table 5.1: Comparison of good homography rate with advanced pass rate, pass rate based on effective match count and original pass rate for three different features and two alternative model estimation methods. We report the percentage of image pairs within the thresholds defined by each measure.

5.3 Results

Finally, we evaluate how well advanced pass rate is able to predict the results of homography estimation. For this we again consider DENSE/RootSIFT, DENSE/LATCH and Relocalized Dense CNN, and compare their result rates of good homography estimation with advanced pass rate, pass rate based on effective match count, and original pass rate. Table 5.1 displays all results. These confirm that original pass rate (right column) highly overrates actual homography estimation performance. Pass rate based on effective match count (5th column) reduces this gap. Yet, only advanced pass rate (4th column) is able to predict the rate of good homography estimation (3rd column) adequately with an average differences of only 2.6%, compared to 9.9% for pass rate based on effective match count. Consequently, together with precision the established effective match distribution score is able to effectively predict homography estimation results.

5.4 Summary and Outlook

In this section we presented a new approach to identify effective matches and developed a score to rank their distribution across an image. Based on this score we establish an advanced pass rate measure and showed its ability to adequately predict the rate of good homography estimation. In the future, this new measure will allow to effectively evaluate different features and filtering approaches for image alignment.

Furthermore, in the context of filtering inliers and outliers with RANSAC, the developed effective match distribution score can be used as a criterion to rank different solutions. However, to evaluate, whether this criterion is more effective than standard inlier count, is a task for future research.

Chapter 6

Performance of Learned Features and Semantic Annotations

In the past handcrafted feature based matching approaches have widely been used for image alignment as well as 3D reconstruction. Their ability to match historic to modern images has been evaluated in Chapter 4, revealing room for improvement. Especially upon the presence of great appearance changes and high occlusion due to structural changes classic handcrafted features lack performance.

A research area closely related to this work is visual localization or place recognition, which focuses on season and illumination but not long time changes [Lowry et al., 2016]. In recent years research in place recognition prospered due to the utilization of Convolutional Neural Networks (CNNs), which are trained for certain recognition tasks. Previous studies show that features extracted from CNNs, pre-trained on large datasets, outperform hand-crafted features especially in challenging conditions such as textureless indoor scenes [Taira et al., 2018] or environments changing appearance across day time and season [Sattler et al., 2018; Sünderhauf et al., 2015b]. Initial approaches regard location recognition as an image retrieval task and only generate global image representations from extracted features [Arandjelovic et al., 2016; Sünderhauf et al., 2015b]. However, more recent works [Sattler et al., 2018; Taira et al., 2018] use features extracted from CNNs to establish one-to-one feature correspondences as required for 6DoF camera pose recognition and exact image alignment.

Another set of studies proposes to exploit semantic scene understanding to enhance location recognition [Garg et al., 2018; Kobyshev et al., 2014; Schönberger et al., 2018; Toft et al., 2018]. These utilize semantic information in different ways. Some works add semantics to feature descriptors to guide the matching process [Kobyshev et al., 2014; Schönberger et al., 2018]. Others use it to prefilter unstable keypoints on mobile objects [Kobyshev et al., 2014] or as a post filter to detect false correspondences [Toft et al., 2018]. Additionally, combinations of these approaches have been presented [Garg et al., 2018].

The performance of methods utilizing pre-trained CNNs and/or semantics has been evaluated in the context of day and night images [Garg et al., 2018; Sattler et al., 2018] and upon the change of seasons [Sattler et al., 2018; Schönberger et al., 2018; Toft et al., 2018], while long term changes of several years have been neglected so far. In this chapter we analyze the suitability of pre-trained CNNs for aligning historic and modern images featuring dramatic structural changes on top of seasonal and illumination change. Besides, we evaluate different approaches utilizing semantic annotations to enhance image matching performance.

6.1 Literature

Place recognition aims at identifying an image's location by comparing it to a large database. This database either consist of a set of images or a 3D point cloud. This leads to two different approaches for place recognition. Structure based methods establish one-to-one feature correspondences between 2D points of an image and a 3D point cloud constructed via SfM. This allows exact 6DoF camera pose estimation of an image. Image based methods on the other hand regard location recognition as an image retrieval task. They establish a global descriptor for an entire image and approximate its location by the most similar image retrieved from a large database.

6.1.1 Place Recognition with pre-trained CNN Features

Initially, place recognition was dominated by local feature-based techniques and bag-of-words approaches [Angeli et al., 2008]. FAB-MAP [Cummins and Newman, 2008] is a popular image based place recognition approach in robotics. It generates bag-of-words representations by utilizing local features such as SIFT [Lowe, 2004] and modeling the cooccurrence probability of image features. In the context of historic to modern image matching we already presented the work of [Fernando et al., 2015], remember Section 3.2.3. They propose bag-of-words representations of densely sampled RootSIFT features to recognize the location of older photographs given modern labeled images from the internet. However, recent studies show, that features generated by Deep Convolutional Neural Networks (CNNs), trained on large datasets, outperform hand-crafted features such as SIFT and SURF.

Sünderhauf et al. [2015b] were the first to show, that CNNs originally trained for object recognition can be utilized in the context of place recognition and are robust against viewpoint and appearance changes. Currently, NetVLAD [Arandjelovic et al., 2016] is one of the state-of-the-art place recognition algorithms based on a CNN combined with a new generalized VLAD layer, trained in an end-to-end manner. Both works belong to the branch of image based location recognition and generate global image representations, while in the context of rephotography we are interested in exact 6DoF pose estimation requiring individual feature correspondences.

Sattler et al. [2018] and Taira et al. [2018] on the other hand follow the structure based approach and extract features from CNNs, regard these as local image descriptors and establish one-to-one feature correspondences between them as in classical image matching. Both show that features extracted from CNNs outperform their hand-crafted counterparts especially in challenging conditions such as textureless indoor scenes [Taira et al., 2018] or environments changing appearance across day time and season [Sattler et al., 2018]. Furthermore, they perform no learning themselves, but extract features from pre-trained CNNs without any fine-tuning. Please note, that training a CNN for the specific task of rephotography image matching is currently not an option, due to a lack of training data.

InLoc: An Indoor Localization Pipeline [Taira et al., 2018]

Taira et al. [2018] use a two step approach to determine 6DoF camera poses in a large scale indoor environment. At first, they use NetVLAD [Arandjelovic et al., 2016], with its pre-trained Pitts30K VGG-16 [Simonyan and Zisserman, 2015] CNN, to retrieve the 100 best matching images. Afterwards, they extract features from conv5 and conv3 of the CNN network for pose estimation. An illustration of the convolutional layers of VGG-16 is presented in Figure 6.1.

In detail Taira et al. [2018] match densely extracted features in a coarse-to-fine manner.

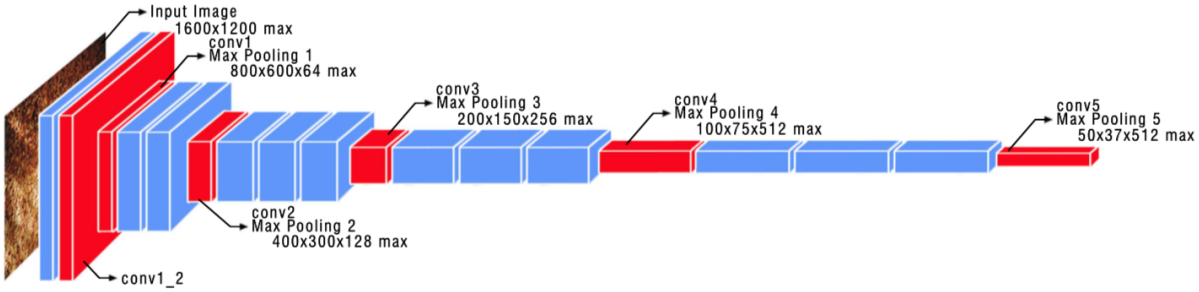


Figure 6.1: Illustration of the convolutional layers of VGG-16 [Simonyan and Zisserman, 2015] as presented in Widya et al. [2018]. Features are extracted from the max pooling layers which have half depth compared to the following layers. For instance, from conv3 features with 256 elements are extracted at a resolution of 1/8 of the original image size.

Initially, they extract conv5 features, with 512 elements at a resolution of only 1/32 of the original image, and match these via mutual nearest neighbour search. Mutual means, that provided a query and a reference image, for a keypoint A in the query image its nearest neighbour B is searched among all keypoints of the reference image. Second, for each keypoint in the reference image its nearest neighbour is searched among all keypoints of the query image. Now, only if A is also the nearest neighbour of B during the second search a mutual correspondence has been detected.

Afterwards, the mutual correspondences found at conv5 restrict the area to search for mutual correspondences between conv3 features, which have a finer resolution of 1/8 of the original image. Finally, homography estimation via RANSAC is used to filter correspondences by geometric consistency. For the top 10 among the 100 retrieved images in terms of RANSAC inlier count, 6DoF query pose is computed with a P3P-RANSAC variant followed by a final pose verification step.

Dense SfM with CNN Features [Sattler et al., 2018]

Sattler et al. [2018] present a benchmark for 6DoF outdoor localization in challenging conditions including day and night as well as seasonal changes. They compare several location recognition approaches, who determine the pose of a query image without prior knowledge of an approximate pose, among others these include NetVLAD and FAB-MAP. Additionally, they propose an optimistic baseline approach, which assumes a small set of retrieved candidate images, similar as in Taira et al. [2018]. Each image of this set is matched to the query image via a DenseSfM method. Again, the VGG-16 network from NetVLAD [Arandjelovic et al., 2016] is applied, but this time the mutual coarse-to-fine matching is performed with conv4 and conv3 features. Finally, correspondences are filtered by estimating up to two homographies between each image pair with RANSAC. Remaining correspondences from all image pairs are used for exact 6DoF pose reconstruction.

Relocalized Dense CNN [Widya et al., 2018]

Widya et al. [2018] adopt the DenseSfM approach of Sattler et al. [2018] for 3D reconstruction and propose a new method to relocalize conv3 features to pixel level accuracy. After a coarse-to-fine matching with conv4 and conv3 features, keypoints from conv3 are relocalized at conv2, conv1 and finally conv1_2, which has the same resolution as the original image, see Figure 6.1.

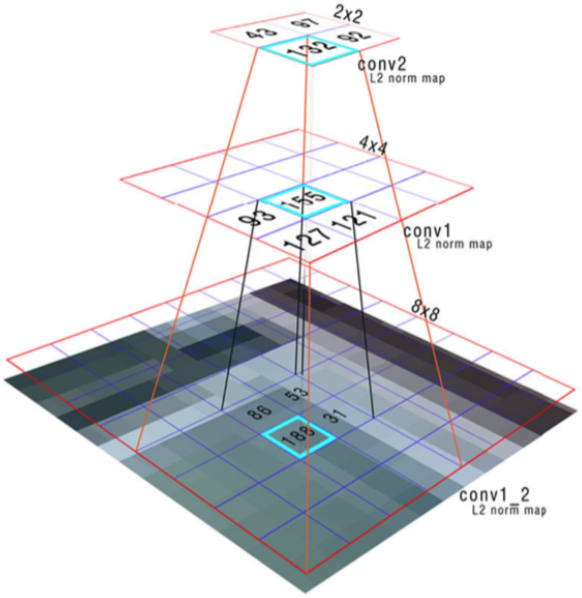


Figure 6.2: Illustration of keypoint relocalization as presented in Widya et al. [2018]. For sparser convolutional layers conv2, conv1 and conv1_2 the L2 norm of each feature is computed. Afterwards, each keypoint is relocalized at the position of the next lower layer which features the maximum L2 value.

Each keypoint at convolutional layer i is relocalized at layer $i-1$ within its corresponding $2x2$ neighborhood. For this the L2 norm of each feature at layer $i-1$ is computed and the keypoint is positioned at the pixel with maximum L2 value, see Figure 6.2. As conv1_2 is reached correspondences have pixel-level accuracy, instead of $8x8$ pixel imprecision. This allows more accurate reconstruction.

CNN-based Local Features

In recent years a variety of local feature descriptors, not hand-crafted such as SIFT or SURF, but learned via CNNs has been proposed as well. These include pure descriptors such as TFeat [Balntas et al., 2016] and DeepDesc [Simo-Serra et al., 2015], but also LIFT [Yi et al., 2016] and DELF [Noh et al., 2017], which couple learning of keypoint extraction and description. However, the evaluation in Chapter 4 and related works [Fernando et al., 2015; Torii et al., 2015] showed that in the presence of strong appearance changes a dense sampling of keypoints is preferable to local feature detection. Furthermore, Schönberger et al. [2017] revealed that learned features are often over-fitted to certain datasets and that advanced hand-crafted features such as DSP-SIFT [Dong and Soatto, 2015] and SIFT-PCA [Bursuc et al., 2015] perform as well as or even better. Thus, we refrain from evaluating any local descriptors learned via CNNs in this work.

Further Approaches to Place Recognition

Please note that there are a variety of further approaches to improve place recognition, which are not relevant for this work. These include sequence based methods, such as SeqSLAM [Milford and Wyeth, 2012], which perform localization based on a sequence of correctly ordered images instead of a single view. However, in rephotography there is usually only a single historic image that needs to be aligned to the modern scene, so sequence based methods are not an option.

Another common idea to handle strong appearance changes is to collect several views of a scene under different conditions, for example at different times of day or at different seasons. As a result, query images can be matched to similar conditions. In case one is not able to collect this additional image data, one may also learn how scenes change across season [Neubert et al., 2015] or day time [Anoosheh et al., 2019; Lowry et al., 2014] and generate additional artificial views. Yet, the structural changes a scene experiences across time spans of several years as present in rephotography, are not predictable as easily.

6.1.2 Place Recognition utilizing Semantics

With the progress of deep learning methods, research in semantic scene understanding prospered [Krizhevsky et al., 2012]. These days, semantic segmentations of images can be generated automatically via pre-trained CNNs such as RefineNet [Lin et al., 2017]. This provides new possibilities to guide feature matching. In the literature there are several examples utilizing semantics to support feature matching and location recognition.

Kobyshev et al. [2014] present a new descriptor based on the semantic context of a keypoint. With this they exclude unstable keypoints, on pedestrians, cars or vegetation before matching. Furthermore, they restrict the set of potential correspondences for each keypoint to semantically close ones. This speeds up the matching process and reduces the number of false correspondences.

Toft et al. [2018] store semantic information for each keypoint in a 3D point model. However, they do not use this information during match generation, but for scoring the reliability of individual 2D-3D correspondences. Thus, they utilize semantic information for outlier detection among previously generated correspondences. In an earlier work Toft et al. [2017] went even further and establish 6DoF camera poses based solely on pixelwise semantic labels. For this they generate a sparse 3D model containing semantically labeled points and curves using only 20 different classes [Cordts et al., 2016]. Afterwards, 6DoF camera pose is optimized by minimizing an error function accessing the projection of the model into the query image.

Mousavian et al. [2015] use semantic labels to guide feature weighting of bag-of-words representations for image retrieval. Schönberger et al. [2018] present a whole semantic localization pipeline for place recognition under extreme appearance (season, time of day, weather) and viewpoint changes ($0° - 180°$). Their main contribution is a new approach to train 3D descriptors encoding high-level geometric as well as semantic information, based on semantic scene completion.

LoST-X [Garg et al., 2018]

Garg et al. [2018] aim at image retrieval of opposite viewpoints. For this they use dense convolutional layers extracted from RefineNet [Lin et al., 2017], a semantic segmentation network trained on the Cityscapes Dataset [Cordts et al., 2016], to establish a novel global image descriptor, named LoST, for image retrieval. In a second step, they use a weighted keypoint matching for the top 10 retrieved images to determine the best match. For an overview of their pipeline, see Figure 6.3.

Their approach to keypoint extraction and matching is entirely different from all those presented previously. In detail they determine the maximally activated region of each conv5 feature map (total of $D = 2048$ maps) and regard such as a keypoint. The match of this keypoint is the maximally activated region in the corresponding feature map D of the reference image. Thus, a fixed number of 2048 keypoint correspondences, at a resolution of 1/32 of the original image, is generated for each image pair. Afterwards, for each of these correspondences the consistency of

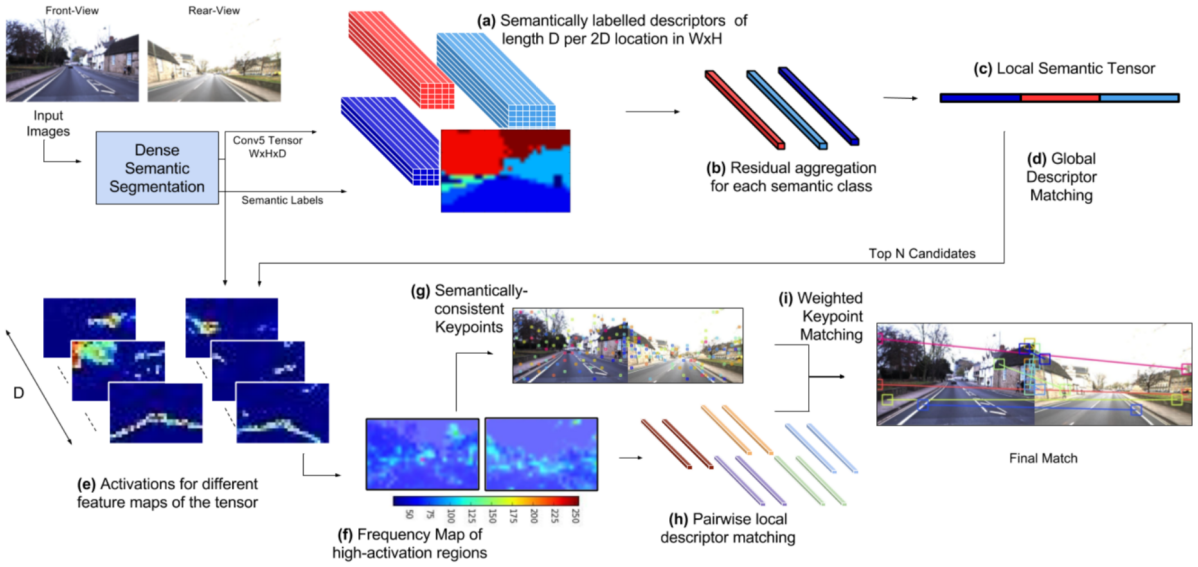


Figure 6.3: Illustration of the LoST-X approach as presented in Garg et al. [2018]. The steps for computing the global image descriptor are depicted at the top (a - c), while the weighted keypoint matching is illustrated in the bottom row (e - i).

semantic labels is checked and only those with the same label are kept. Finally, each correspondence is weighted by descriptor distance to establish an overall matching score for each image pair. The image with the highest score is the final match retrieved.

6.2 Methods

In the following we describe all elements of the feature matching pipelines compared in this evaluation. These include different approaches to feature extraction, feature matching and local as well as global correspondence filters. Furthermore, we introduce different semantic filters used for keypoint reduction prior to matching. For evaluation we use the nyc-grid dataset presented in Section 4.2 and our previous work [Becker and Vornberger, 2019], while the main evaluation criteria are *precision*, *effective match distribution* (Chapter 5) and *good homography rate* (Section 4.3.5).

6.2.1 Features

In the following we briefly describe all evaluated features. If not stated differently keypoints are matched via direct nearest neighborhood search.

Classic Detectors: As a representative of a classic local feature approach we include upright RootSIFT [Arandjelović and Zisserman, 2012] in our evaluation. Furthermore, we compare densely sampled keypoints described by RootSIFT and LATCH [Levi and Hassner, 2016], who showed the highest performance in the previous evaluation (Chapter 4 and [Becker and Vornberger, 2019]).

Relocalized Dense CNN: We follow the approach of Sattler et al. [2018] and pass images to a state-of-the-art CNN network, choosing the same VGG-16 network [Simonyan and Zisserman, 2015] as NetVLAD [Arandjelovic et al., 2016]. Now, we extract dense features of conv4 and

conv3 for each image. Furthermore, we use the relocalization approach presented by Widya et al. [2018] to establish an exact pixel position for each conv3 feature. In accordance with Sattler et al. [2018] and Widya et al. [2018] we perform a mutual coarse-to-fine matching with conv4 and conv3 features, to generate pixel-level correspondences. Please note that with this approach despite mutual matching a single keypoint may belong to more than one correspondence. This is a result of coarse-to-fine matching between conv4 and conv3, which for each keypoint on conv4 also takes the areas of neighboring keypoints on conv3 into account.

Relocalized Dense CNN conv3: Additionally, we only extract features from conv3, relocalize these to pixel positions and match them directly via nearest neighbour search across the entire image.

LoST-X Keypoints: We were inspired by the novel approach of keypoint correspondence extraction proposed by Garg et al. [2018]. Thus, we want to evaluate whether the individual keypoint correspondences generated by this approach are not only suitable for image retrieval, but also image alignment as required in rephotography. To do so we pass all images through RefineNet [Lin et al., 2017] for semantic segmentation and feature generation. Afterwards, we use the same approach as Garg et al. [2018] to extract keypoint correspondences and filter these by semantic label consistency. For utilizing individual keypoint correspondences, opposed to aggregating them for a global matching score, we need to position them at individual pixels. To achieve this, we again use the method of Widya et al. [2018] and relocalize LoST-X Keypoints extracted from conv5, to exact pixel positions on the original image.

6.2.2 Semantic Category Assignment

In the following we describe several keypoint and correspondence filters, that we compare during our evaluation. Many of these are based on semantics and require the assignment of a semantic category to each keypoint. To generate these semantic categories we pass all images through RefineNet [Lin et al., 2017], a state of the art semantic segmentation network, which assigns a semantic category to each image pixel. Each keypoint receives the semantic category of the pixel it is located at. We use the network configuration, generated by training on the Cityscapes Dataset [Cordts et al., 2016], that distinguishes between 20 classes. These we assign to 6 groups for our evaluation, see Table 6.1.

Group	**Semantic Categories**
mobile objects	person, rider, car, truck, bus, train, motorcycle, bicycle
man made structures	building, wall, fence, pole, traffic light, traffic sign
vegetation	vegetation
ground	road, sidewalk, terrain
sky	sky
void	void

Table 6.1: Overview of semantic categories and their grouping.

6.2.3 Prefilters

Inspired by Kobyshev et al. [2014] we implement two semantic prefilters, which are supposed to filter unstable keypoints before the matching process.

Semantic Prefilter Mobile: This excludes all keypoints on mobile objects, including people and vehicles, from the matching process.

Semantic Prefilter Mobile + Vegetation: This excludes all keypoints on mobile objects and additionally removes keypoints on vegetation above ground level, primarily trees and some bushes. We regard keypoints on the these as unstable, since most vegetation regenerates itself every year and experiences even greater change across the long time spans faced in rephotography.

6.2.4 Semantically Guided Nearest Neighbour Matching

Along the lines of Kobyshev et al. [2014] we guide the matching process by limiting potential matches to semantically similar keypoints.

Weak Semantic Matching: We allow a matching to keypoints that belong to the same semantic group, remember Table 6.1. This means a keypoint on a wall can also be matched to one belonging to a building or a traffic sign.

Strict Semantic Matching: We limit potential matches to keypoints with identical category. As a consequence, a keypoint on a building can only be matched to another building and one on a wall to another wall.

6.2.5 Local Correspondence Filters

Due to dense feature extraction and great appearance changes between image pairs, direct matching results in many false correspondences. These need to be filtered, yet classic methods such as the ratio test [Lowe, 2004] or a maximally descriptor distance threshold are not suitable for this task, as our previous study [Becker and Vornberger, 2019] showed. Instead, we apply the following local correspondence filters.

Mutual Matching: Initial results revealed that in practice especially many-to-one correspondences are an issue during homography estimation. Hence, we evaluate the application of mutual matching, commonly used with CNN features [Sattler et al., 2018; Sünderhauf et al., 2015b; Widya et al., 2018].

Many-To-One Filter (DGS or descriptor distance based): If mutual matching is applied only one-to-one correspondences remain, though mutual matching is far more restrictive. Thus, as an alternative we propose a local filtering mechanism limited to many-to-one correspondences. This compares all correspondences sharing a single keypoint via a certain criterion and only keeps the best correspondence. In our case the criteria tested are descriptor distance and Disparity Gradient Sum (DGS) [Becker and Vornberger, 2019; Roth and Whitehead, 2000].

Semantic Group Consistency: Additionally, we test a weak semantic consistency filter, which eliminates all correspondences whose keypoints do not share the same semantic group.

Semantic Category Consistency: The strict semantic consistency filter on the other hand only tolerates correspondences with exactly the same category.

Filter Mobile Correspondences: This filter eliminates all correspondences that comprise at least one keypoint on a mobile object.

Filter Mobile + Vegetation Correspondences: This approach eliminates correspondences including at least one keypoint on a mobile object or vegetation.

Please note, all these correspondence filters are not exclusive, but can be combined with each other, while not all combinations make sense. Additionally, the effects of the last four semantic filters, may depend on prefilters and matching techniques applied.

6.2.6 Global Correspondence Filters

To further reduce the number of false correspondences we apply diverse geometric correspondence filters, which take the properties of other correspondences into account to detect outliers.

DGF: For comparison to our previous evaluation, see Chapter 4, we apply the disparity gradient filter, initially developed by Roth and Whitehead [2000], and successfully applied in the context of historic to modern image matching by Ali and Whitehead [2014] and in our previous work [Becker and Vornberger, 2019].

DGF + K-VLD: Furthermore, we use the successful combination of DGF followed by K-VLD [Liu and Marlet, 2012], presented in Chapter 4.

RANSAC Homography: As proposed in related works [Sattler et al., 2018; Torii et al., 2015; Widya et al., 2018], we estimate up to 5 homographies via RANSAC [Fischler and Bolles, 1981] and remove all correspondences not fitting any of these.

RANSAC Homography (semantic): Additionally, inspired by Toft et al. [2017], we estimate 5 homographies via RANSAC ranked by semantic consistency instead of inliers. In detail we apply a dense keypoint sampling to the first image and project all keypoint into the second image via the candidate homography. Afterwards, we count the number of keypoints, which have the same semantic category as their projection into the other image.

RANSAC Homography (distribution): Furthermore, we estimate up to 5 homographies via RANSAC ranked by the effective match distribution score (Chapter 5) of their inliers instead of the number of inliers.

6.3 Results

This section is structured as follows. At first, we present the results of LoST-X Keypoints and the standard Relocalized Dense CNN pipeline of Widya et al. [2018] compared to local and dense classical features. Afterwards, we compare the individual filtering approaches proposed in the previous section. Finally, the top performing pipelines for each feature are compared.

6.3.1 Initial Performance Comparison

At first, we evaluate the Relocalized Dense CNN approach of Widya et al. [2018] with mutual coarse-to-fine matching between conv4 and conv3 and estimation of 5 homographies via RANSAC. Additionally, we test a direct mutual matching of conv3 features. These approaches we compare to a selection of local and densely sampled classical features, again applying mutual matching and a global RANSAC Homography filter. Furthermore, we present the results of Dense LATCH and Dense RootSIFT, with simple nearest neighbour matching and DGF application followed by K-VLD if LATCH descriptor is used. These two approaches showed the best performance in the previous evaluation from Chapter 4. Finally, the performance of LoST-X Keypoints is accessed.

Since the final goal is homography estimation, we compare all these approaches by their ability to estimate accurate homographies for image transformation. In detail we define three

different accuracy levels for an estimated homography. At the first level an estimated transformation satisfies the weak criterion (mean distance < 60, area scale factor > 0.9), remember Section 4.3.5. These estimations contribute to the percentage of good homographies. Furthermore, we classify more accurate transformations satisfying the strong criterion (mean distance < 30, area scale factor > 0.95) as very good homographies. Additionally, we define a class of excellent homographies with mean distance < 15 and area scale > 0.97.

Table 6.2 shows the results for all these approaches after homography estimation via RANSAC [Fischler and Bolles, 1981] and LMedS [Rousseeuw, 1984]. Results confirm that dense sampling clearly outperforms local features. Furthermore, Relocalized Dense CNN outperforms Dense LATCH and Dense RootSIFT in combination with mutual matching and RANSAC Homography filtering. Additionally, mutual coarse-to-fine matching between conv4 and conv3 shows better performance than direct mutual matching of conv3 features. However, the best performance is achieved by Dense LATCH in combination with DGF and K-VLD and Dense RootSIFT with DGF, as proposed in Chapter 4.

Striking is the substandard performance of LoST-X Keypoints. These demonstrate themselves unsuitable for historic to modern image alignment, despite good results during image retrieval on the entire dataset, see Table 6.3. With the simple LOST descriptor without weighted keypoint matching only 17 of 52 images were successfully retrieved. Instead, if additional weighted keypoint matching was applied to the top 10 matches, 27 images were successfully retrieved. Thus, during image retrieval the keypoint matching approach of Garg et al. [2018] is successful. Yet, the individual keypoint correspondences established are not suitable for image alignment.

	Hom. Estimation RANSAC			Hom. Estimation LMedS		
	Good	Very Good	Excellent	Good	Very Good	Excellent
Reloc.Dense CNN mutual and RANSAC Hom. F.	67.3	42.3	19.2	67.3	44.2	17.3
Reloc.Dense CNN conv3 mutual and RANSAC Hom. F.	59.6	36.5	13.5	55.8	34.6	17.3
Dense LATCH mutual and RANSAC Hom. F.	63.5	42.3	23.1	63.5	36.5	21.2
Dense LATCH DGF+K-VLD	71.2	46.2	23.1	71.2	42.3	25.0
Dense RootSIFT mutual and RANSAC Hom. F.	51.9	25.0	15.4	59.6	32.7	15.4
Dense RootSIFT DGF	67.3	32.7	19.2	71.2	50.0	26.9
upright RootSIFT mutual and RANSAC Hom F.	46.2	34.6	15.4	34.6	28.8	11.5
LoST-X Keypoints	1.9	0.0	0.0	0.0	0.0	0.0

Table 6.2: Results of LoST-X Keypoints and the Relocalized Dense CNN approach of Widya et al. [2018] compared to local and densely sampled classical features using mutual matching and RANSAC Homography filtering. Additionally, the results for Dense LATCH and Dense RootSIFT with simple nearest neighbour matching and DGF followed by K-VLD are provided. The percentage of image pairs aligned within the three accuracies is reported. For each column top values are marked in red, while second best results are marked in blue.

LoST-X	direct	top 2	top 10	all(52)
retrieved images	17	22	27	37

Table 6.3: Performance of LoST-X for image retrieval on the entire dataset. Modern images are used as the reference set and for each historic image the closest match is queried. Using the simple LOST descriptor only 17 images are retrieved correctly. Instead, if additional weighted keypoint matching is used for the top matches far more images are successfully retrieved.

We mainly attribute this to two factors. First, during image retrieval all individual keypoint correspondences contribute to a global score. This global score might be sufficient for comparing potential image matches, which show high variation in scenery, even though most individual correspondences are not very accurate. Second, the resolution, of only 1/32 of the original image, at which descriptors are extracted is very low. Such high layer region matching is effective in image retrieval, but can not be sampled down to actual keypoints at pixel level. In comparison CNN features of conv3 are extracted at a resolution of 1/8 of the original image.

Therefore, LoST-X Keypoints are not considered in the rest of the evaluation. Instead, we test whether altering the standard CNN pipeline of Widya et al. [2018] and Sattler et al. [2018] leads to performance improvements. Furthermore, we evaluate the effects of utilizing semantic annotations for keypoint matching and correspondence.

6.3.2 Global Correspondence Filters

During matching rephotographs many false correspondences are a major challenge. Thus, correspondence filtering is an essential step for successful homography estimation. Furthermore, initial results revealed great performance differences between application of DGF and RANSAC Homography filtering for densely sampled features. Hence, at first we compare different approaches for global correspondence filtering, including estimation of up to five homographies via RANSAC [Torii et al., 2015; Widya et al., 2018], the Disparity Gradient Filter (DGF) [Becker and Vornberger, 2019; Roth and Whitehead, 2000] and DGF followed by K-VLD [Liu and Marlet, 2012]. Additionally, we evaluate two alternative RANSAC Homography filtering approaches, ranking homographies via their semantic fit or effective match distribution score, as explained in Section 6.2.6. Please note that no further filters are applied during this comparison. This means that for all features direct nearest neighborhood matching is performed, these includes CNN features from conv3. Only for Relocalized Dense CNN mutual coarse-to-fine matching is performed between conv4 and conv3.

Figure 6.4 compares the performance of all five correspondence filters in terms of good homography rate after model estimation with RANSAC and LMedS. Both graphs clearly illustrate that DGF and DGF + K-VLD outperform all three RANSAC Homography Filters, while DGF slightly outperforms its combination with K-VLD for all features apart from Dense LATCH. The great performance gap between DGF and RANSAC Homography filtering for Relocalized Dense CNN with conv3 features and all classic feature matching approaches confirms that DGF is better suited for handling large numbers of outliers. Instead, if additional filtering is performed, such as mutual matching in the case of Relocalized Dense CNN, the difference in performance is less significant. Yet, even for mutually matched CNN features DGF achieves higher good homography estimation rates.

A comparison between the different RANSAC Homography filtering approaches shows that standard ranking by inlier count is most effective in terms of good homography estimation. On the other hand, ranking homographies by effective match distribution leads to comparable

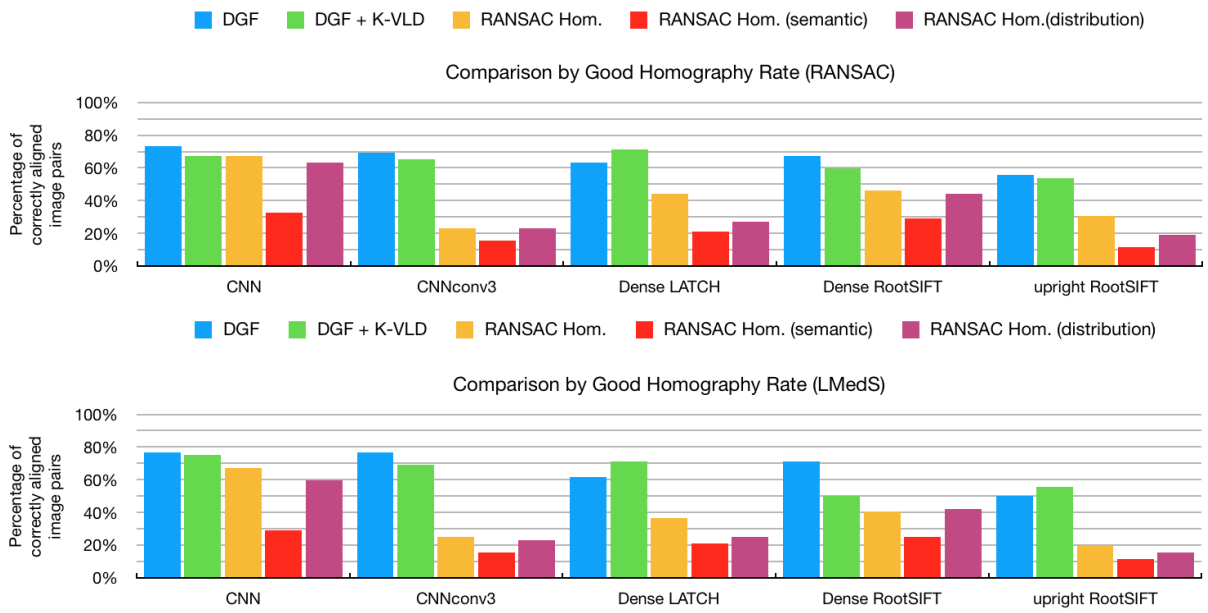


Figure 6.4: Comparison of different global correspondence filters via good homography rate after model estimation with RANSAC (top) and LMedS (bottom).

results for both CNN approaches as well as Dense RootSIFT, but shows worse performance for Dense LATCH and upright RootSIFT. Ranking homographies by semantic fit on the other hand results in the lowest performance across all features. We mainly attribute this to the great changes in scenery occurring across time, even if only the mentioned semantic categories are considered. Across years often scene parts are covered by growing or entirely new vegetation. Furthermore, demolished and new buildings change the silhouettes of semantic categorization, see Figure 6.5. Additionally, the dataset contains several close-ups of single buildings or street sections, in which no sky or vegetation is visible at all. The majority of pixels of these images belong to a single category thus no silhouettes useful for image alignment are formed.

Thus, we conclude that accurate camera relocalization purely based on semantics, as proposed by Toft et al. [2017], might be an option for mostly static scenes suffering from seasonal but not greater structural changes. Similarly, mostly static scenes have been considered in the related works of Kobyshev et al. [2014] and Schönberger et al. [2018]. However, if facing severe structural changes as they are common in rephotography, matching based on semantic silhouettes is not an option.

In general, exclusive DGF application outperforms DGF followed by K-VLD. Only in the case of DENSE LATCH additional K-VLD application increases performance in terms of good homography rate. As described in our previous evaluation (Section 4.3.4), K-VLD is able to boost precision, but it does so at the cost of discarding several correct correspondences. As a result, additional K-VLD application is only useful if after previous filter application precision values are still to low for accurate model estimation. This is the case for Dense LATCH, but not for other features.

In the rest of this evaluation we combine all approaches to evaluate with correspondence filtering via DGF and DGF followed by K-VLD for Dense LATCH. This allows us to compare all evaluated matching methods and further filters by their influence on final model estimation. Furthermore, in the case of Dense LATCH we additionally monitor whether the application of additional filters makes K-VLD application unnecessary.

Figure 6.5: Examples of image pairs were pure semantic alignment fails. In the left image across time large parts of the image are occupied by new vegetation. Additionally, the two towers on the left have been replaced by a new building with an entirely different silhouette. Instead, the right scene is almost completely covered by a building, so that the majority of image pixels feature the same semantic category.

6.3.3 Mutual Matching vs. Many-To-One Filter

Second, we evaluate the effectiveness of three local correspondence filters not requiring semantic information. These include mutual matching and the two proposed many-to-one filters utilizing descriptor distance or Disparity Gradient Sum (DGS) for correspondence elimination. We evaluate, whether these filters have a positive influence on model estimation, compared to simple direct nearest neighbour matching. Please note that we do not alter the matching pipeline of standard Relocalized CNN with mutual coarse-to-fine matching between conv3 and conv4. In this case refraining from mutual matching would increase the number of potential correspondences too much. Yet, with CNN features directly extracted from conv3 we are able to access the effects on CNN features.

Figure 6.6 analyses the effect of different local filters in terms of Effective Match Distribution. In detail, the top graph shows the percentage of images with an effective match distribution score above 35, which was determined as the required value for successful model estimation in Section 5, while the bottom graph displays the average effective match distribution score across the entire dataset. Both graphs illustrate, that mutual matching is the most restrictive filter and eliminates not only false correspondences, but also reduces the number of correct matches significantly, resulting in lower values of effective match distribution. These lower scores are not only visible, when focusing on average values (bottom), but also reduce the relevant percentage of images featuring a score above 35. Instead, for the many-to-one filters the reduction of image pairs with effective match distribution score above 35 is approximately half of that induced by mutual matching. The effects on precision are in the opposite direction. This means mutual matching results in the most significant increase in precision, while many-to-one filters boost precision by about half. Furthermore, a comparison between both many-to-one filters shows that the criterion used for elimination has hardly any effect.

However, finally we are interested whether any local filter is able to improve the percentage of images for which good homography estimation is achieved. Thus, as mentioned previously we additionally apply DGF and DGF followed by K-VLD for Dense LATCH to allow homography estimation for final image alignment. Figure 6.7 displays the good homography rates achieved with mutual matching, many-to-one filters and simple nearest neighbour matching. Results show that all evaluated filters hardly have any effect on the percentage of good homographies. Instead, results without applying any filters show the highest or very similar performance.

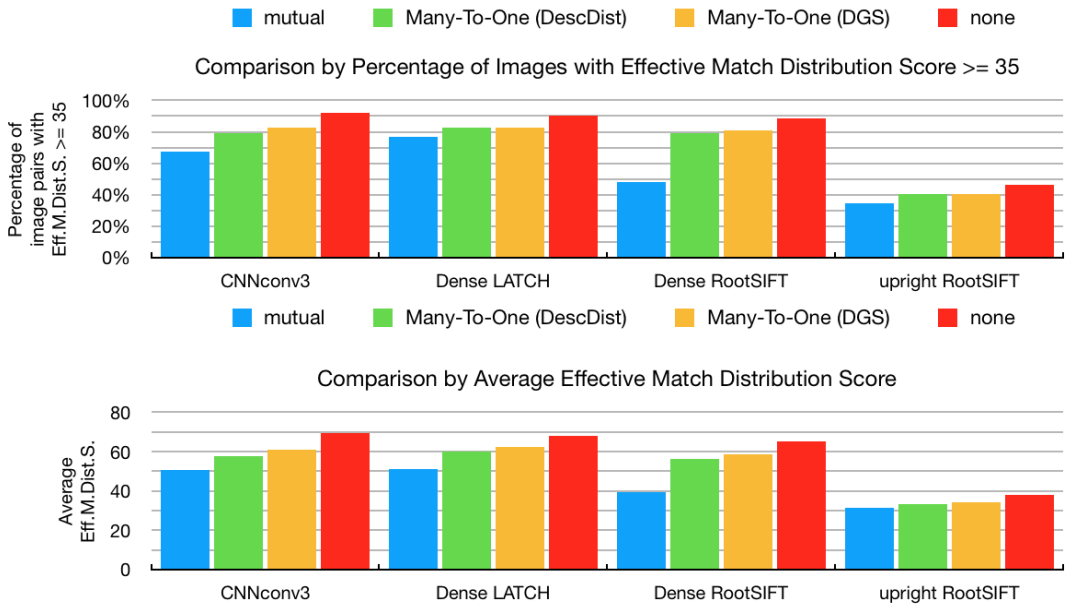


Figure 6.6: Effects of mutual matching and Many-To-One filtering, with correspondence elimination based on descriptor distance and Disparity Gradient Sum (DGS), in terms of effective match distribution score. The top graph compares the percentage of images with an effective match distribution score above 35, while the bottom one displays the average score across all image pairs of the dataset.

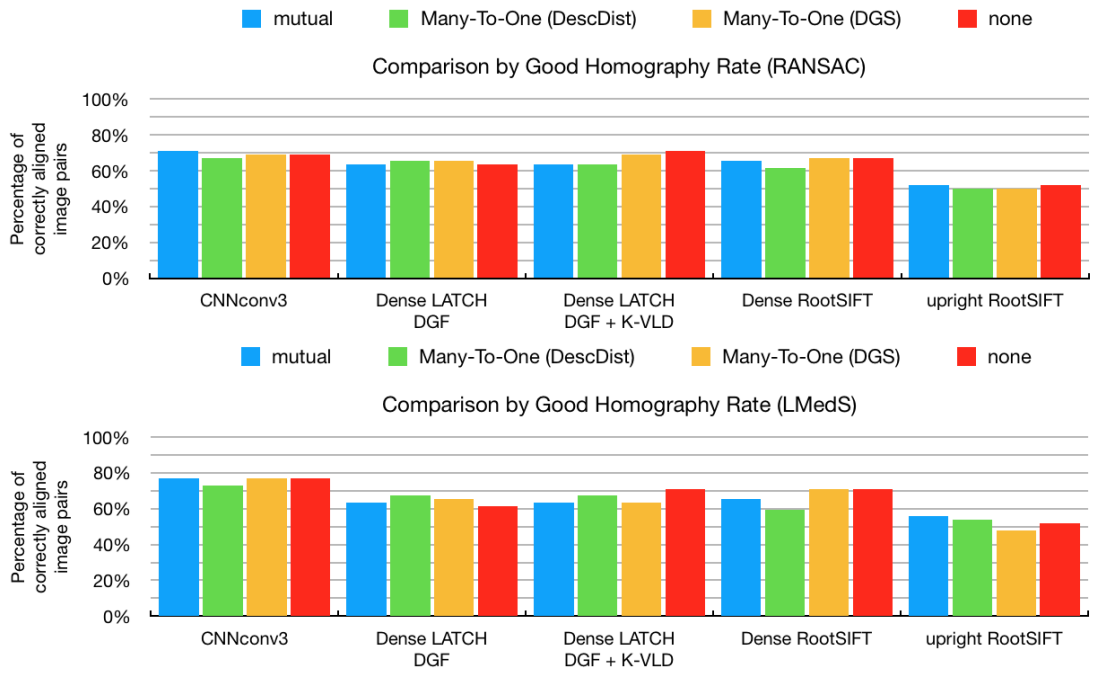


Figure 6.7: Effects of mutual matching and Many-To-One filtering, with correspondence elimination based on descriptor distance and Disparity Gradient Sum (DGS), on good homography rate after model estimation via RANSAC (top) and LMeds (bottom).

This indicates that DGF is well suited to cope with many-to-one matches and requires no additional local filters handling many-to-one correspondences. We mainly attribute this to the following. In general, two kinds of many-to-one correspondences exist. In the first case many-to-one matches are very close to each other, so that they are usually not regarded as outliers during model estimation. Since model estimation is in general not that accurate, these kind of many-to-one matches often support correct model estimation. Additionally, their disparity gradient values are close by. Consequently, DGF filtering profits from these. In the second case many-to-one matches are distributed far across the entire image. Hence, their disparity gradients vary a lot and DGF commonly eliminates these false correspondences anyway.

6.3.4 Semantically Guided Matching

Next, we compare two semantically guided matching approaches with global direct nearest neighbour matching. As explained previously strict semantic matching, limits potential correspondences to those keypoints featuring the same semantic category, while weak semantic matching allows correspondences within semantic groups. Figure 6.8 displays the homography estimation results for all three matching approaches. Again, we apply DGF for all features and DGF followed by K-VLD for Dense LATCH before model estimation with RANSAC (left) and LMedS (right), with no further filtering mechanisms used. Furthermore, we do not perform semantically guided matching for Relocalized Dense CNN, since features on conv4 usually already incorporate some semantic information [Sünderhauf et al., 2015a]. Thus, with mutual coarse-to-fine matching between conv4 and conv3, some form of semantic filtering already takes place.

Results show that for CNN features from conv3 and DENSE LATCH, simple direct nearest neighbour matching outperforms or performs equal to both semantic matching approaches. In the case of Dense RootSIFT weak semantically guided matching shows a little higher performance than simple matching, yet the difference is marginal. Instead, for upright RootSIFT strict semantically guided matching is best, at least if LMedS is used for model estimation, otherwise performance is very similar for all three approaches. Overall, semantically guided matching does not lead to significant improvements in terms of final model estimation.

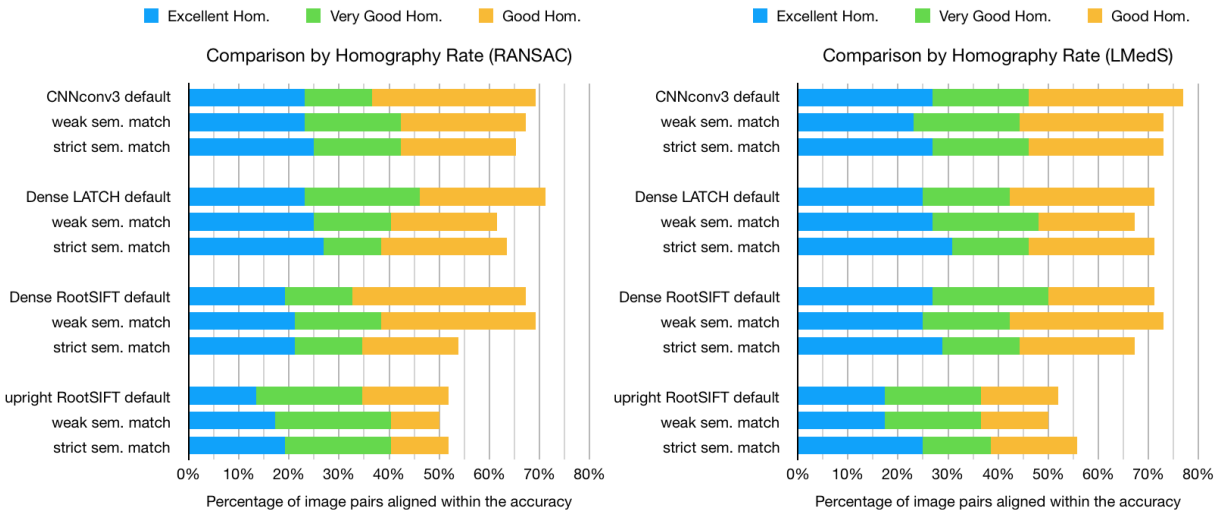


Figure 6.8: Performance comparison of weak and strict semantically guided against simple direct nearest neighbour matching based on the achieved accuracy of homography estimation.

We mainly attribute this to the following reasons. First of all, as mentioned previously, there are some images in the dataset whose pixels almost entirely belong to a single semantic category, remember Figure 6.5. For these no effective restriction of potential matches takes place if semantic matching is used. For image pairs with more semantic categories on the other hand, restricting matches to a single semantic category does not significantly improve the number of valuable correspondences or the ratio between correct and false correspondences. This is the case, since commonly matches who feature similar descriptors and therefore are potential correspondences for a single keypoint lie close to each other and belong to the same semantic category. A popular example are repetitive patterns on building facades. Consequently, semantically guided matching might be a little faster due to the reduction of possible matches, yet it does not significantly improve performance or matching accuracy.

6.3.5 Semantic Prefilters

An alternative to a semantically guided matching process is to eliminate keypoints, considered to be unstable, before the standard matching approach. Here we compare two semantic prefilters which eliminate keypoints on mobile objects only or on both mobile objects and vegetation. Afterwards, direct nearest neighborhood matching is performed with all remaining keypoints. As previously we apply DGF for all features and DGF + K-VLD for Dense LATCH before model estimation with RANSAC and LMedS, with no further filtering mechanisms used. Please note, that we do not apply a semantic prefilter for Relocalized Dense CNN with coarse-to-fine matching between conv4 and conv3. This is, since conv4 features matched at first step span an area of 16x16 pixels of the original image that often features different semantic categories. Thus, the decision which keypoints to eliminate is more complex than for the simple semantic prefilter we propose and results would no longer be comparable to those of the other features.

Figure 6.9 shows homography estimation results for both prefilters. In general, the effects of prefilter application vary across features as well as model estimation approaches. Considering model estimation via RANSAC (left), prefilter application improves performance for CNN features from conv3 and Dense LATCH with DGF only. Yet, for Dense LATCH with DGF followed by K-VLD as well as combinations including RootSIFT, prefilter application leads to very

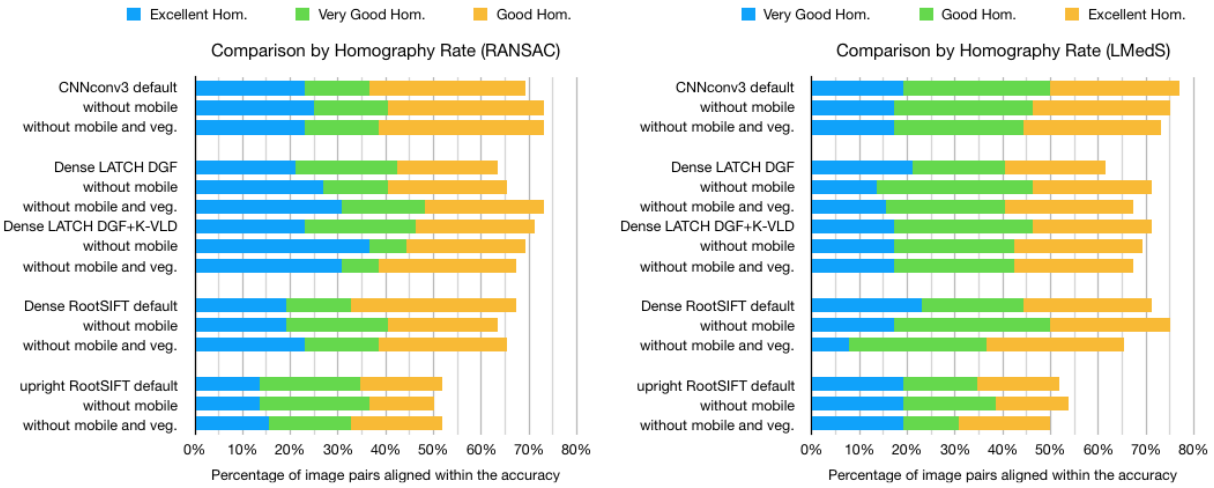


Figure 6.9: Effects of semantic prefilters, eliminating keypoints on mobile objects and vegetation, in terms of the accuracy of homography estimation.

similar or even slightly lower performance. Instead, if model estimation via LMedS (right) is performed, prefilter application only shows a clear positive effect for Dense LATCH with DGF. For RootSIFT only eliminating keypoints on mobile objects leads to better performance, while for CNNconv3 features performance slightly decreases.

Consequently, it is clear that applying a prefilter does not result in significant performance improvements across features. Yet, there is also no significant harm to performance when a mobile prefilter is applied. Especially Dense LATCH profits from eliminating keypoints on mobile objects, since this restricts the number of false correspondences to a critical value, that allows to omit K-VLD application, without any performance losses.

Otherwise we attribute these results to the following. Correspondences involving keypoints on correctly classified mobile objects have to be false, since during the time spans between rephotographic image pairs mobile objects are not repeatedly captured. Yet, these false correspondences seem to be successfully identified by DGF anyway, thus keypoint prefiltering does not result in large performance improvements. On the other hand, since keypoints of mobile objects can not result in valuable correct correspondences, eliminating these does not harm performance either.

In the context of keypoints on vegetation this does not necessarily apply. In general vegetation renews itself throughout the seasons. Furthermore, across the long time spans present in rephotography vegetation either grows massively or is regularly replaced. Thus, keypoints on vegetation are usually not repeated across image pairs. However, in practice classification of vegetation is not as accurate as that of mobile objects. Especially on older photographs, dark areas on buildings are often misclassified as vegetation. Furthermore, trees regularly lead to the classification of a whole treetop area as vegetation, even if the image was taken in winter and trees do not have any foliage. Both issues are illustrated in Figure 6.10. Thus, if all keypoints from regions classified as vegetation are eliminated, this often includes keypoints on buildings, which have a high chance of repetition across years. As results show, under these circumstances, the elimination of keypoints on vegetation regularly harms model estimation.

Figure 6.10: Illustration of current issues with semantic segmentation of vegetation. For comparison we show the original image together with its corresponding semantic segmentation as created by RefineNet [Lin et al., 2017]. Vegetation is marked in light green. The left example illustrates the misclassification of dark areas, on buildings, as vegetation, while the right shows the common output produced if trees are present in an image. This shows that even if it is winter and trees do not feature foliage, great areas around their branches are marked as vegetation, even though they show the building behind the trees.

		Hom. Estimation RANSAC			Hom. Estimation LMedS		
		Good	Very Good	Excellent	Good	Very Good	Excellent
Reloc.Dense CNN	none	73.1	46.2	21.2	76.9	46.2	25.0
	group consistency	73.1	55.8	28.8	76.9	51.9	26.9
	category consistency	67.3	42.3	25.0	73.1	50.0	25.0
Reloc.Dense CNN conv3	none	69.2	36.5	23.1	76.9	46.2	26.9
	group consistency	69.2	44.2	19.2	78.8	48.1	26.9
	category consistency	69.2	36.5	23.1	76.9	50.0	23.1
Dense LATCH DGF	none	63.5	42.3	21.2	61.5	42.3	21.2
	group consistency	63.5	44.2	25.0	67.3	48.1	30.8
	category consistency	63.5	42.3	28.8	69.2	48.1	25.0
Dense LATCH DGF+K-VLD	none	71.2	46.2	23.1	71.2	42.3	25.0
	group consistency	73.1	44.2	21.2	69.2	46.2	26.9
	category consistency	59.6	44.2	23.1	67.3	48.1	30.8
Dense RootSIFT	none	67.3	32.7	19.2	71.2	50.0	26.9
	group consistency	65.4	44.2	23.1	67.3	42.3	23.1
	category consistency	63.5	38.5	23.1	69.2	42.3	25.0
upright RootSIFT	none	51.9	34.6	13.5	51.9	36.5	17.3
	group consistency	46.2	36.5	19.2	50.0	36.5	21.2
	category consistency	50.0	36.5	19.2	50.0	34.6	19.2

Table 6.4: Comparison of filtering correspondences by semantic group and category consistency in terms of achieved homography estimation accuracy. The percentage of image pairs aligned within the three accuracies is reported. Top values for each column of a single feature are marked in red, while top values for each column across features are underlined.

6.3.6 Semantic Consistency

Next, we evaluate the potential of utilizing semantic information for correspondence filtering. For this we refrain from applying any prefilter or semantically guided matching. Instead, we use standard direct nearest neighbour matching for all keypoints and try to identify false correspondences resulting from this process via semantics. Afterwards, we apply DGF (followed by K-VLD for Dense LATCH) before model estimation with RANSAC and LMedS.

At first, we compare the effects of enforcing group or category consistency between correspondences. This means all correspondences whose keypoints do not belong to the same semantic group or category are eliminated. Table 6.4 displays the performance with and without correspondence filter application in terms of final homography estimation. For most features model estimation via LMedS produces improved results compared to RANSAC. Considering the results for LMedS, apart from RootSIFT, all features profit from enforcing semantic consistency. In the cases of Relocalized Dense CNN and DENSE LATCH demanding group or category consistency of correspondences particularly improves model estimation upon high accuracies. Furthermore, in line with prefilter application, for Dense LATCH, filtering correspondences by semantic consistency allows to avoid K-VLD application, without great performance losses. On the other hand, whether group consistency or category consistency should be preferred is not that clear, yet in terms of very good to excellent accuracy, at least for CNN features and Dense LATCH, group consistency shows a slightly higher performance.

Second, we compare the effects of removing correspondences involving keypoints on mobile objects and vegetation. Table 6.5 displays the performance with and without filtering mobile correspondences only and mobile plus vegetation correspondences. No semantic consistency of correspondences was demanded upon the generation of these results. Again, model estimation with LMedS generally results in higher performance. Mobile and vegetation correspondence

		Hom. Estimation RANSAC			Hom. Estimation LMedS		
		Good	Very Good	Excellent	Good	Very Good	Excellent
Reloc.Dense CNN	none	73.1	46.2	21.2	76.9	46.2	25.0
	remove mobile corresp.	71.2	44.2	25.0	78.8	40.4	23.1
	mobile and vegetation	69.2	44.2	23.1	78.8	48.1	26.9
Reloc.Dense CNN conv3	none	69.2	36.5	23.1	76.9	46.2	26.9
	remove mobile corresp.	73.1	38.5	23.1	75.0	46.2	28.8
	mobile and vegetation	71.2	36.5	21.2	73.1	46.2	28.8
Dense LATCH DGF	none	63.5	42.3	21.2	61.5	42.3	21.2
	remove mobile corresp.	67.3	42.3	23.1	67.3	38.5	25.0
	mobile and vegetation	65.4	42.3	23.1	65.4	42.3	26.9
Dense LATCH DGF+K-VLD	none	71.2	46.2	23.1	71.2	42.3	25.0
	remove mobile corresp.	69.2	42.3	25.0	67.3	44.2	25.0
	mobile and vegetation	63.5	40.4	26.9	69.2	46.2	23.1
Dense RootSIFT	none	67.3	32.7	19.2	71.2	50.0	26.9
	remove mobile corresp.	65.4	42.3	17.3	67.3	48.1	23.1
	mobile and vegetation	61.5	42.3	17.3	63.5	42.3	25.0
upright RootSIFT	none	51.9	34.6	13.5	51.9	36.5	17.3
	remove mobile corresp.	48.1	38.5	21.2	44.2	30.8	17.3
	mobile and vegetation	51.9	40.4	23.1	46.2	34.6	13.5

Table 6.5: Effects of eliminating correspondences featuring keypoints on mobile objects, compared to eliminating correspondences with keypoints on mobile objects and vegetation. The percentage of image pairs aligned within the three accuracies is reported. Top values for each column of a single feature are marked in red, while top values for each column across features are underlined.

filtering shows slightly positive effects in terms of homography estimation for LATCH and CNN features. Yet, these are less significant than those of enforcing group consistency. Thus, we assume that DGF already filters out most false correspondences including keypoints on mobile objects or vegetation.

6.3.7 Combined Performance

At last, we combine the most successful semantic correspondence and prefilters for each feature, to compare performance across features after optimizing the entire matching pipeline. Table 6.6 displays the performance results of selected matching pipelines for each feature, in terms of homography estimation accuracy.

In general, model estimation via LMedS outperforms model estimation via RANSAC. Especially with semantic correspondence filtering LMedS shows improved performance for very good and excellent accuracy. This shows that semantic correspondence filtering methods as well as DGF succeed in boosting precision for the majority of image pairs, so that additional outlier detection during model estimation is no longer required. Once this is the case, model estimation via LMedS is favorable, since it considers all remaining correspondences and is more likely to output an adequate transformation of the entire image. RANSAC instead, might only consider a minority of correspondences resulting in a model with low reprojection error but only fitting to a small image area.

Furthermore, results show that for all features apart from Dense RootSIFT additional semantic correspondence or prefilters improve performance especially in terms of model accuracy. Yet, which exact combination of filters is best varies across features and often several filter combinations show comparable results. For instance in the case of Dense LATCH with DGF there is no great difference between enforcing group or category consistency in combination with a mo-

		Hom. Estimation RANSAC			Hom. Estimation LMedS		
		Good	Very Good	Excellent	Good	Very Good	Excellent
Reloc.Dense CNN	none	73.1	46.2	21.2	76.9	46.2	25.0
	group consistency	73.1	55.8	28.8	76.9	51.9	26.9
	group consistency + remove mobile and veg.	76.9	48.1	30.8	78.8	51.9	32.7
Reloc.Dense CNN conv3	none	69.2	36.5	23.1	76.9	46.2	26.9
	group consistency	69.2	44.2	19.2	78.8	48.1	26.9
	group consistency + remove mobile corresp.	73.1	42.3	23.1	76.9	50.0	26.9
	group consistency + prefilter mobile	69.2	36.5	21.2	73.1	50.0	23.1
Dense LATCH DGF	none	63.5	42.3	21.2	61.5	42.3	21.2
	group consistency + remove mobile and veg.	67.3	42.3	28.8	67.3	44.2	28.8
	group consistency + prefilter mobile and veg.	71.2	44.2	28.8	69.2	44.2	30.8
	category consistency + remove mobile and veg.	71.2	38.5	26.9	67.3	50.0	30.8
	category consistency + prefilter mobile and veg.	63.5	46.2	23.1	63.5	46.2	30.8
Dense LATCH DGF+K-VLD	none	71.2	46.2	23.1	71.2	42.3	25.0
	group consistency + remove mobile and veg.	69.2	46.2	26.9	69.2	46.2	28.8
	group consistency + prefilter mobile and veg.	71.2	46.2	26.9	71.2	51.9	26.9
Dense RootSIFT	none	67.3	32.7	19.2	71.2	50.0	26.9
	category consistency + remove mobile and veg.	65.4	42.3	21.2	63.5	48.1	25.0
upright RootSIFT	none	51.9	34.6	13.5	51.9	36.5	17.3
	category consistency + prefilter mobile and veg.	50.0	38.5	26.9	55.8	42.3	26.9

Table 6.6: Results for combinations of different semantic correspondence and prefilters, to allow comparison of optimized matching pipelines for each feature. The percentage of image pairs aligned within the three accuracies is reported. Top values for each column of a single feature are marked in red, while top values for each column across features are underlined.

bile and vegetation prefilter or removing correspondences involving keypoints on mobile objects and vegetation. However, it is clear that with additional semantic filtering Dense LATCH DGF reaches performances comparable to those of Dense LATCH with DGF followed by K-VLD. Thus, additional K-VLD application can be avoided.

Overall, Relocalized Dense CNN with coarse-to-fine matching between conv4 and conv3 outperforms all other features if group consistency of correspondences is required and correspondences between mobile objects and vegetation are eliminated, as shown by the underlined values in Table 6.6. The second best performance is achieved by Relocalized Dense CNN features directly extracted from conv3. Considering the lower computational cost at which these can be extracted and matched, their use is preferable for many applications. Furthermore, semantic prefiltering is easily applicable to CNN features from conv3, which is able to further speed up the matching process. The computational cost of semantic segmentation is also relevant. Yet, if no semantic correspondence or prefiltering is employed at all, the performance of Relocalized Dense CNN features with coarse-to-fine matching between conv4 and conv3 is almost equal to those of CNN features directly extracted from conv3.

Considering non CNN features, Dense LATCH and Dense RootSIFT achieve similar performances in terms of final homography estimation. Notably, in the case of DENSE RootSIFT this performance is achieved without any additional semantic filtering, while in the case of Dense LATCH, at least in terms of model accuracy additional semantic filtering improves performance.

6.4 Summary

In this chapter we analyzed the suitability of features extracted from CNNs, pre-trained for location recognition, for historic and modern image matching. Besides, we evaluated different approaches utilizing semantic annotations to enhance matching performance. In summary, our evaluation revealed the following.

At first, densely extracted features from CNNs pre-trained for location recognition can be successfully applied in the context of modern to historic image matching and outperform handcrafted features. Additionally, results reveal that coarse-to-fine matching between two convolutional layers does not improve performance significantly compared to direct extraction of features from the finer layer. Yet, even though the pre-trained CNN features used in this evaluation outperform all other evaluated approaches, potential for improvement still remains, since only 80% of rephotographs from the nyc-grid dataset are successfully aligned and for only 50% of images a very good accuracy is reached.

Second, in the context of rephotography DGF clearly outperforms RANSAC in false correspondence identification. This is because DGF is able to correctly identify outliers even if facing inlier ratios of only 1%, as reported in Chapter 4 and Becker and Vornberger [2019]. Instead, RANSAC requires too many iterations to successfully handle such low inlier ratios.

Third, for model estimation LMedS is preferable to RANSAC, as soon as precision values have reached a certain level. This is since LMedS considers all remaining correspondences and thus succeeds in computing a global image transformation for the majority of image pairs. RANSAC instead, is more likely to compute a local transformation well fitting a small part of the image, but not representing the desired global image transformation.

Finally, semantic segmentation is able to support alignment of historic and modern images, yet currently its improvements on performance are not dominant and mostly model accuracy is increased. Consequently, since additional semantic segmentation leads to a noticeable increase in computational cost, one needs to decide for each individual application, whether such is justifiable.

6.4.1 Discussion

All conclusions were drawn after investigation of features from a single pre-trained CNN, namely the Pitts30K VGG-16 from Arandjelovic et al. [2016]. Furthermore, only the effects of semantic annotations produced by RefineNet [Lin et al., 2017] were evaluated. Consequently, the results of this evaluation mainly provide an indication how well pre-trained CNN features and semantic annotations are suited for matching historic and modern images. At the time of execution of this study we selected both algorithms, since they represented state of the art approaches in their respective domain of location recognition and semantic segmentation. However, if different pre-trained CNNs or upcoming more advanced semantic segmentation algorithms are utilized results may vary.

In the following we discuss the shortcomings of the applied approaches and make suggestions how to improve future performance in the context of rephotography.

Pre-trained CNN Features for Rephotography

The Pitts30K VGG-16 network from NetVLAD [Arandjelovic et al., 2016], has been trained with 10.000 images from Google Street View Time Machine depicting several places at different times. The goal of Arandjelovic et al. [2016] was to establish an image representation invariant to changes in viewpoint, illumination and season. Furthermore, the network needs to handle partial occlusions and learn to ignore unstable objects such as people and vehicles. In general these requirements are very common to those of image descriptors in the context of rephotography. Yet, during modern to historic image matching one faces even more challenges. At first, larger occlusions are common and secondly historic images often feature a low quality compared to modern high resolution digital images.

Learned features tend to generalize less well than their hand-crafted counterparts [Schönberger et al., 2017]. Thus, adapting the training data to the specific problem faced is the most promising approach to improve performance of CNN features in the context of historic to modern image matching. Consequently, we suggest to use NetVLAD to train an alternative VGG-16 network better adapted to the images present in rephotography. For this the Pitts30K dataset should be expanded by several scenes for which historic as well as modern images are available.

The magnitude of 30.000 images contained in the original dataset provides an idea of how many historic images are required for this approach. We already explained that the dataset in this thesis only contains 52 images, since it is very difficult to acquire rephotographs of a scene taken from the exact same viewpoint. Yet, for training the NetVLAD architecture images do not need to share the exact same viewpoint. Instead, we suggest to use a search engine to collect images from popular sights across different periods of time as done on a small scale by Ali and Whitehead [2014]. These sights usually do not change as rapidly across time as the images of the nyc-grid dataset, yet they feature a similar low quality. Furthermore, the training dataset could be extended by rephotographs collected on the re.photos portal (more in Chapter 7).

Another observation is that all evaluated features are not able to handle view limitations caused by fences occupying large parts of a scene commonly present during constructions. Humans instead are able to ignore distracting fences and concentrate on the scene behind these. Thus, we suggest that a network needs to be trained, which similar to mobile objects is able to ignore view limiting fences. To achieve this there are two options. First, one could add a significant amount of scenes which have undergone construction recently and add images of the scene with and without fences to the training dataset. Alternatively, artificial fences can be added to images already present in the dataset and additionally supply these fenced images for training.

Semantic Segmentation for Rephotography

RefineNet [Lin et al., 2017] is a network for dense semantic segmentation of images. This work utilized its variant trained on the Cityscapes Dataset [Cordts et al., 2016], containing approximately 3000 urban images. During the application of RefineNet two major drawbacks could be observed in the context of rephotography.

At first, precise semantic segmentation often fails for historic images. In detail, traffic signs, traffic lights and poles these are placed on are successfully identified in modern images, while in older images with lower resolution and fading colors these are regularly not detected. Furthermore, in historic images dark areas of buildings, suffering from poor illumination, tend to be misclassified as vegetation. An example of both of these observations is shown in Figure 6.11. Second, trees lead to a marking of large areas as vegetation, even if it is winter and they do not

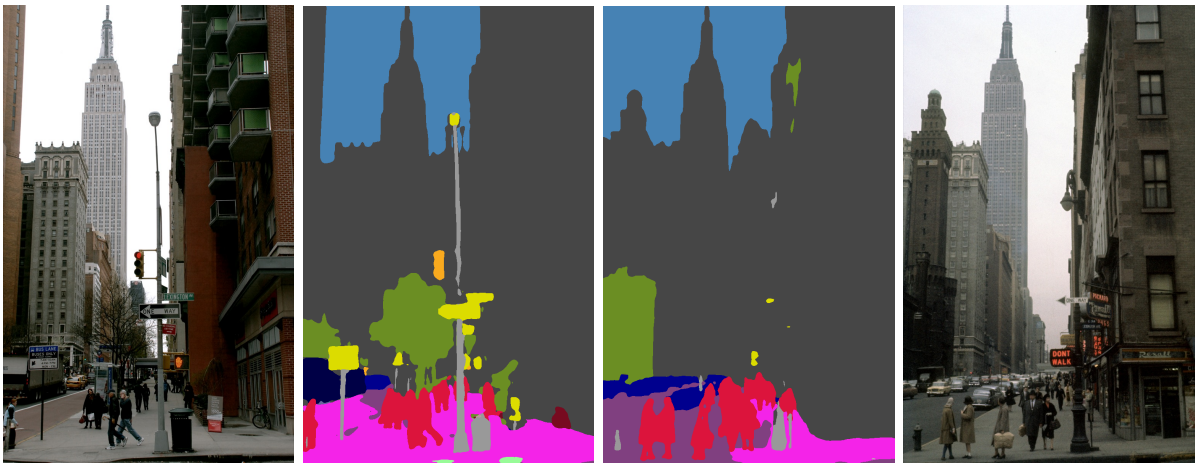


Figure 6.11: Semantic categorization of RefineNet for an image pair from the nyc-grid dataset. On the left, the modern image and its segmentation are displayed while on the right the old images segmentation is followed by the original image. For modern images categorization is very detailed and features clear edges, especially for road signs and vegetation. Instead, in older images with lower resolution and fading colors road signs are often not detected and dark areas are misinterpreted as vegetation. An example of this is shown in the bottom left part of the right segmentation. Original images by Paul Sahner, NYC-Grid.

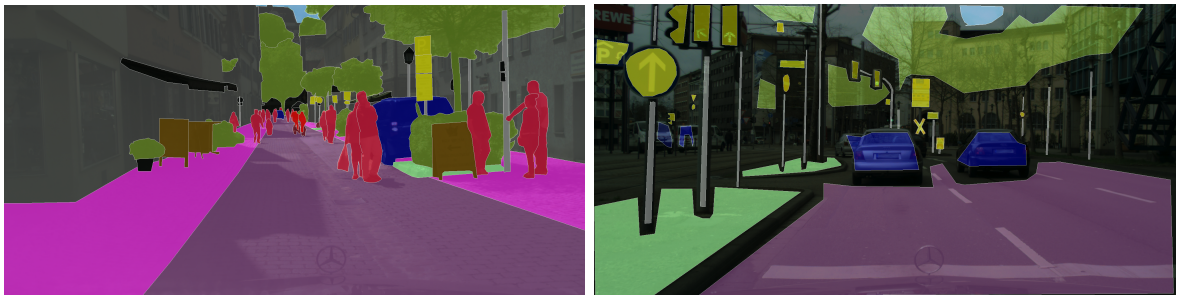


Figure 6.12: Two examples of images from the Cityscapes Dataset [Cordts et al., 2016], with their semantic classes encoded by different color overlays. The left image features a dense pixel annotation and belongs to the set of images with fine annotations, while the right image shows an example of coarse annotation.

have any foliage. Consequently, informative areas between tree branches, mostly belonging to buildings, loose their correct semantic classification. Please remember, this issue was already mentioned in the context of semantic prefiltering and illustrated in Figure 6.10.

Both of these drawbacks can be explained by the composition of the Cityscapes Dataset used for training. At first, this only contains high resolution images of different modern cities. Thus, its failure to adequately segment historic photographs of lower quality is not surprising. Furthermore, the dataset provides 5000 images with fine annotations and an additional set of 20000 images with coarse annotations. One example of each is presented in Figure 6.12. Both examples illustrate, that the ground truth labeling of trees provided as training data exactly replicates the observed output. This is, even trees without foliage are marked as if they feature a crown covered with leaves, so that large parts of the background are marked as vegetation This approach might be useful when comparing images across seasons, but is not favorable in the context of rephotography.

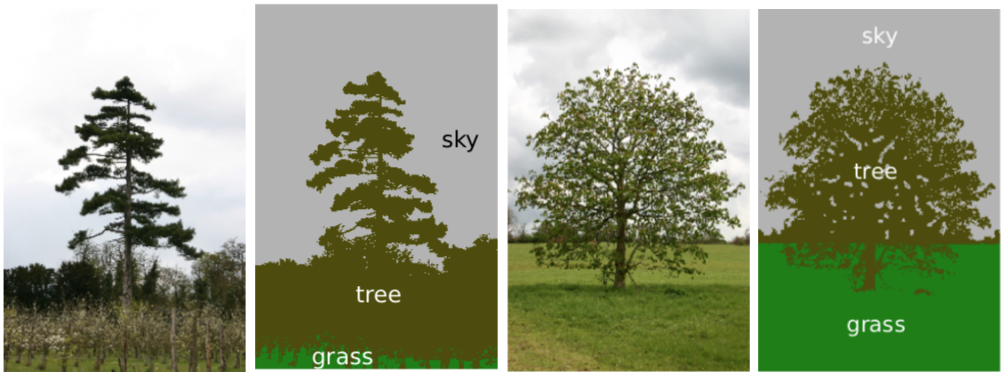


Figure 6.13: Illustration of a ground truth semantic labeling of trees as it is desirable in rephotography. Images from Krähenbühl and Koltun [2011].

Consequently, if one is interested in utilizing dense semantic segmentation for rephotography alignment in the future, we suggest to start with creating a new dataset for training networks such as RefineNet. Next to modern high resolution images, this should contain diverse historic photographs of urban scenes containing adequately labeled traffic signs, poles and fences. Furthermore, in all images vegetation, especially trees, should be annotated in a fine manner. This means, if background pixels are visible in between leaves or branches of trees these need to receive the semantic label of the background. Two examples of the desired ground truth fine tree labeling are presented in Figure 6.13. However, please note that creating such a new dataset with ground truth semantic annotations is time consuming and its positive impact on aligning rephotographs is not guaranteed.

6.4.2 Outlook

For future work it is desirable to construct an additional dataset containing modern and historic images for which ground truth 6DoF camera poses are available. To create this, 3D models of several modern scenes can be constructed via SfM and one or more historic images can be aligned to each model via manually selected point correspondences. Thus, ground truth poses between the historic image and all images of the modern scene are generated and can be added to the dataset.

Such a dataset can be utilized to perform more reliable comparisons between current and future approaches for registering historic and modern image pairs. Furthermore, the suitability of algorithms aimed at guiding the user to the viewpoint of a reference photograph, could be investigated in a more sophisticated way.

Direct Camera Pose Estimation from CNNs [Melekhov et al., 2017]

Once such a dataset with ground truth 6DoF camera poses exists, this might be used to train a CNN for direct pose estimation as in Melekhov et al. [2017].

Melekhov et al. [2017] trained a CNN, which takes two images as an input and directly estimates the relative pose (rotation and translation) between two cameras. They successfully apply the CNN to challenging image pairs showing repetitive structures, textureless scenes, large viewpoint and illumination changes and compare its performance against pose estimation approaches based on classic feature detectors such as SURF [Bay et al., 2006] and ORB [Rublee et al., 2011]. Overall, the CNN's pose estimation results are comparable to those of classic approaches. How-

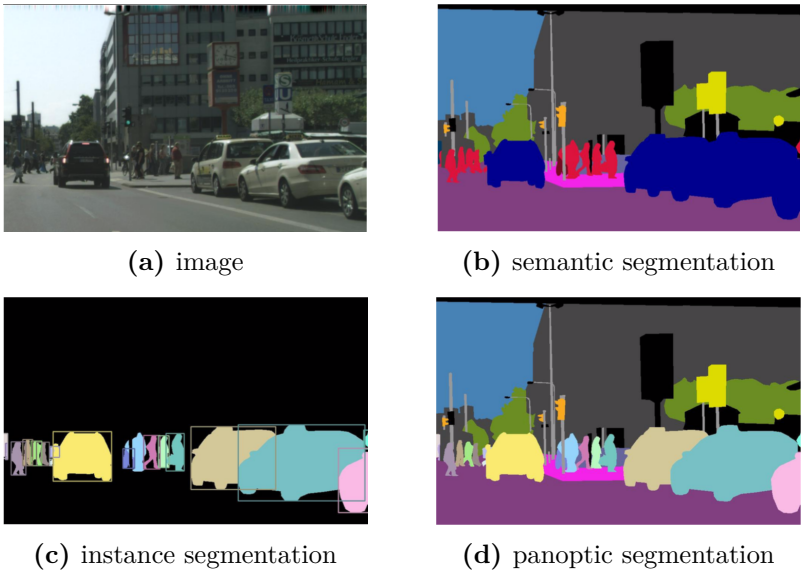
ever, in more challenging scenes, were texture is missing or viewpoint changes are too large to identify enough feature matches, the CNN still adequately predicts relative poses and thus outperforms classic feature detection.

It would be interesting to evaluate such a direct approach in the context of historic to modern image matching. Since the CNN trained by Melekhov et al. [2017] is able to estimate relative pose, despite missing texture and large viewpoint changes, a CNN might also be able to handle structural and appearance changes occurring across long time periods. However, due to the different nature of the task, the training of a new CNN is necessary. Melekhov et al. [2017] trained their CNN on the DTU dataset [Jensen et al., 2014], which contains images of 80 scenes captured from 49 or 64 different positions recorded under 7 different lightning conditions. This provides a general idea, how big the training dataset should be.

Furthermore, a CNN trained for relative pose estimation, could be used for pose recovery during image capture, but not be directly utilized for post-processing. Instead, for post-processing training a second CNN targeted at homography estimation is required.

Panoptic Segmentation

Another field of research, which gained attention very recently, is panoptic segmentation [Kirillov et al., 2019]. Panoptic segmentation combines the two well established challenges of semantic segmentation and instance segmentation. Semantic segmentation aims at assigning a class label to each pixel of an image, while instance segmentation detects and segments individual objects present in an image. The combined panoptic segmentation allows to assign different classes to all contents of an image and at the same time identify individual objects of each class, see Figure 6.14.



(a) image **(b)** semantic segmentation

(c) instance segmentation **(d)** panoptic segmentation

Figure 6.14: Different segmentations of image (a). (b) shows a semantic segmentation where a class label is assigned to each pixel. (c) shows an instance segmentation, where different instances of objects (cars and people) are identified. In (d) both approaches are combined to a panoptic segmentation, where each pixel receives a class as well as an instance label. Images from [Kirillov et al., 2019]

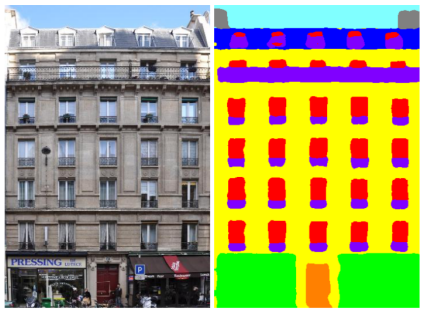


Figure 6.15: Detailed segmentation of a building facade with classes such as window, door and balcony as shown in Liu et al. [2017].

The field of panoptic segmentation is still in its infancy and the panoptic segmentation as proposed in Figure 6.14 is not very useful in the context of historic to modern image matching. This is, since objects such as cars and pedestrians do not repeat themselves across long time periods. Thus, they do not need to be associated with other instances of the same class in a second image. However, a more comprehensive panoptic segmentation including different instances of buildings, road signs and trees is desirable in the context of rephotography. Especially a more detailed segmentation of facades, as performed in Liu et al. [2017] and shown in Figure 6.15, in combination with instance identification could be valuable.

A thorough segmentation of building instances could be used to match individual buildings against each other and determine those, who are still present in a modern scene opposed to those who disappeared. In this context the bounding boxes of buildings and windows generated during instance segmentation, might provide the key information to identify buildings, whose facades received a make over, but whose general shape did not change across years. This could lead to new approaches for historic to modern image matching as well as 6DoF pose recovery of historic images.

However, no algorithm performing such a segmentation of building instances has been presented so far, nor does a dataset exist, which identifies individual instances of buildings in images, outside the domain of satellite imagery analysis. Thus, it is still a long way till this kind of segmentation may be utilized in the context of modern to historic image alignment. Nonetheless, future work should keep an eye on new developments in this field of research.

6.4.3 Conclusion

This chapter proved the general applicability of features extracted from mid-level layers of CNNs, originally trained for location recognition, for aligning historic and modern images in rephotography. Currently, location recognition in general, and location recognition via deep learning is a popular field of research and new approaches are presented every year [Anoosheh et al., 2019; Sarlin et al., 2019]. Thus, future research in the area of rephotography alignment should pay attention to these developments and monitor which approaches are applicable to further improve modern to historic image matching.

Next, we suggest to create a new dataset containing historic images of low resolution and with fading colors in addition to modern high resolution and quality digital images. Training location recognition networks on this dataset will improve the robustness of features to appearance changes across long time periods. More robust features are also desirable for related disciplines such as matching night-to-day images or matching images across seasons. Furthermore, a dataset

containing historic and modern images accompanied by ground truth 6DoF camera pose can be used to perform more reliable comparisons between current and future approaches for registering rephotographs.

Besides, we showed that additional semantic information is able to improve the alignment of rephotographs especially in terms of accuracy. Semantic segmentation is also a fast growing field of research, with regular presentations of new approaches such as training semantic segmentation on synthetic data [Chen et al., 2019], as well as improved semantic segmentation networks [Liu et al., 2019]. Consequently, we recommend to keep an eye on developments in this area as well.

Chapter 7

Practical Developments in Computationally Assisted Rephotography

The motivation for this thesis is to develop methods that assist in the creation, post-processing and presentation of rephotographs. While the previous chapters have focused on the scientific fundamentals and methods required for this, this part of the thesis focuses on practical approaches and challenges faced in developing tools to support rephotographers.

At first, an online platform, developed at the University of Osnabrueck, for organizing and presenting rephotographs from all over the world, is presented. Following is a discussion on the challenges users face during post processing their rephotographs and present ideas to simplify this process. Second, several mobile applications are introduced, that support recovering the exact view point of a photograph, and the current shortcomings of these applications are discussed.

Please note, the works presented in this chapter have been developed by bachelor and master students of the University of Osnabrueck supervised by or in regular consultation with the author of this thesis.

7.1 Re.Photos

At the beginning of this work, we realized several projects as well as individuals use rephotography to document all kinds of changes across time. Upon others this includes works showing urban development [Klett et al., 1984; Levere et al., 2004], visualizing glacier melting [Gore, 2006; usgs.gov] and environmental changes[1]. Furthermore, several photographers perform rephotography as a hobby and publish their pictures on personal websites[2]. Consequently, online a variety of publicly available rephotographs exists but they are rather difficult to find.

To change this, we developed an online portal, which allows users to upload, register and present their own rephotographs. The initial implementation was realized within the scope of a bachelor thesis by Weber [2015]. Besides, the portal was recently presented to the scientific community in Schaffland et al. [2019].

[1] `http://tilbakeblikk.no`, `http://denalirepeatphotos.uaf.edu`, `http://repeatphotography.org` (all accessed on January 7th, 2020)

[2] `http://nyc-grid.com` (accessed on April 30th, 2017), `http://asakinneyproject.blogspot.com` (accessed on January 30th, 2016)

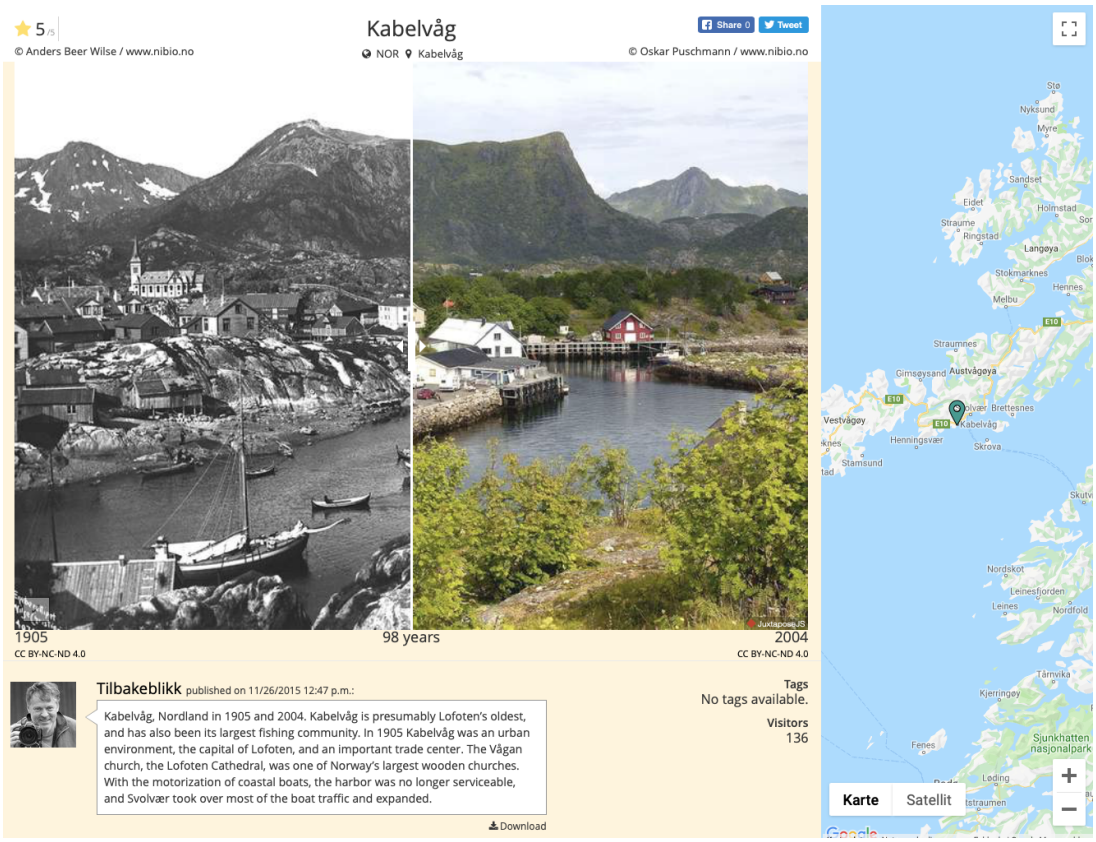


Figure 7.1: Presentation of a rephotographic compilation on the online portal re.photos. The slider, currently in the center of both images, can be moved to the left and right to reveal the entire historic and modern photograph.

Presentation

Figure 7.1 illustrates the presentation of a rephotographic image pair on the portal. Both photographs are displayed on top of each other and a slider can be moved across the images to reveal the entire historic or modern shot. At the corners the names of the photographers and the time each image was taken as well as the copyright information is displayed, while on the right side the scenes location is shown in a map. Furthermore, authors can add a description to their images and users can rate rephotographs and post comments.

Rephotographers

Registered users are able to upload images to the platform, register these (see Section 7.1.1) and publish them. Besides, the images itself this process requires additional data from the user to allow presentation and sorting of compositions as shown in Figure 7.1 and 7.2. To ease the upload process for rephotographers as much information as possible is extracted from the EXIF data of the uploaded images. This includes the date and time of image capture as well as the location. Other information such as the names of the photographers and the image description have to be supplied by the author. Yet, often for old images information on the location and capture time has to be supplied by the author, due to a lack of EXIF data.

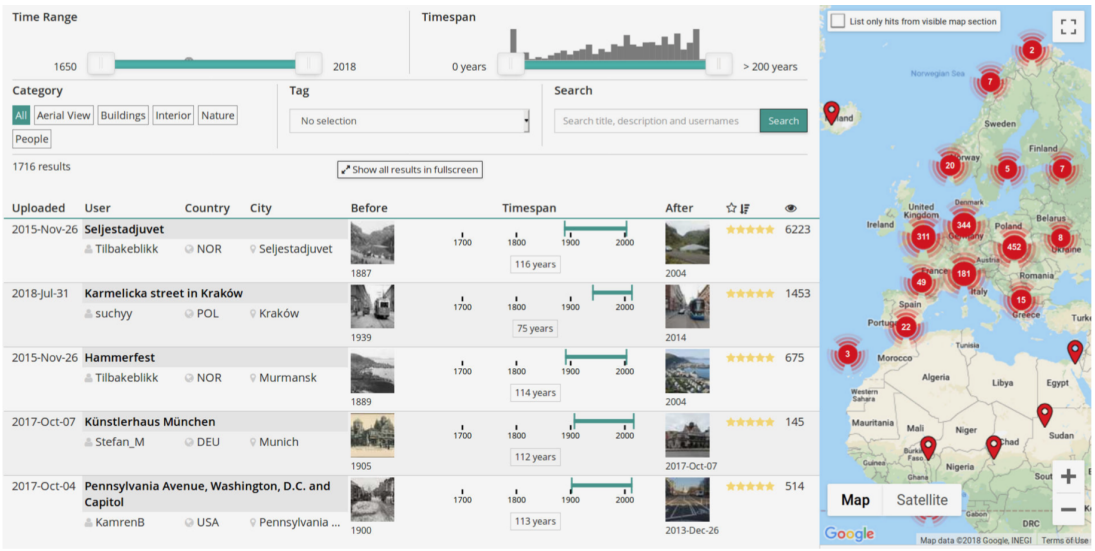


Figure 7.2: Illustration of the standard browsing view of re.photos. Image from [Schaffland et al., 2019].

The Public

Every visitor of the website is able to browse through all presented rephotographs. For this process image pairs can be selected by several parameters including scene type, time range of the historic image or time span between images. Search parameters such as title, description or author can be utilized as well. Resulting rephotographic pairs are presented in a table, which itself can be sorted by upload time, time span or user ranking, see also Figure 7.2.

An alternative option to browse image pairs is by location. For this a map is displayed on the right side of the browsing window. This contains a mark for each image pair, while markings are accumulated if the region viewed on the map contains many photographs. Clicking on accumulation markers leads to a scrolling into regions until individual image pairs can be selected. Furthermore, the map can be used to limit the rephotographs listed in the table by their visibility on the current map section.

More recently the option to browse images by tags has been added to the platform. In this context a new view was created, in which previews of several rephotographs belonging to a single tag are displayed, see Figure 7.3. Thus, the viewer gets an idea of the images belonging to a certain tag and can easily decide whether he is interested in these. This browsing option might be preferable for visitors who are new to the platform, not looking for specific rephotographs and are a little overwhelmed by the variety of available images.

Additionally, registered users are able to rate rephotographs with 1 to 5 stars and post individual comments under each image pair.

Image Archive and Forum

Sometimes it is very difficult to acquire historic photographs. On the other hand, several people posses a bunch of old photographs and might be interested in their rephotographs, but do not want to spent time on creating these. For this reason the portal allows users to upload historic images, accompanied by all information available, to an image archive. Then, interested rephotographers can visit the scene, take a new image and complete the rephotographic pair.

Furthermore, the website provides a forum, at which users can share information about digital image archives, they have found online, as well as general advice on taking rephotographs.

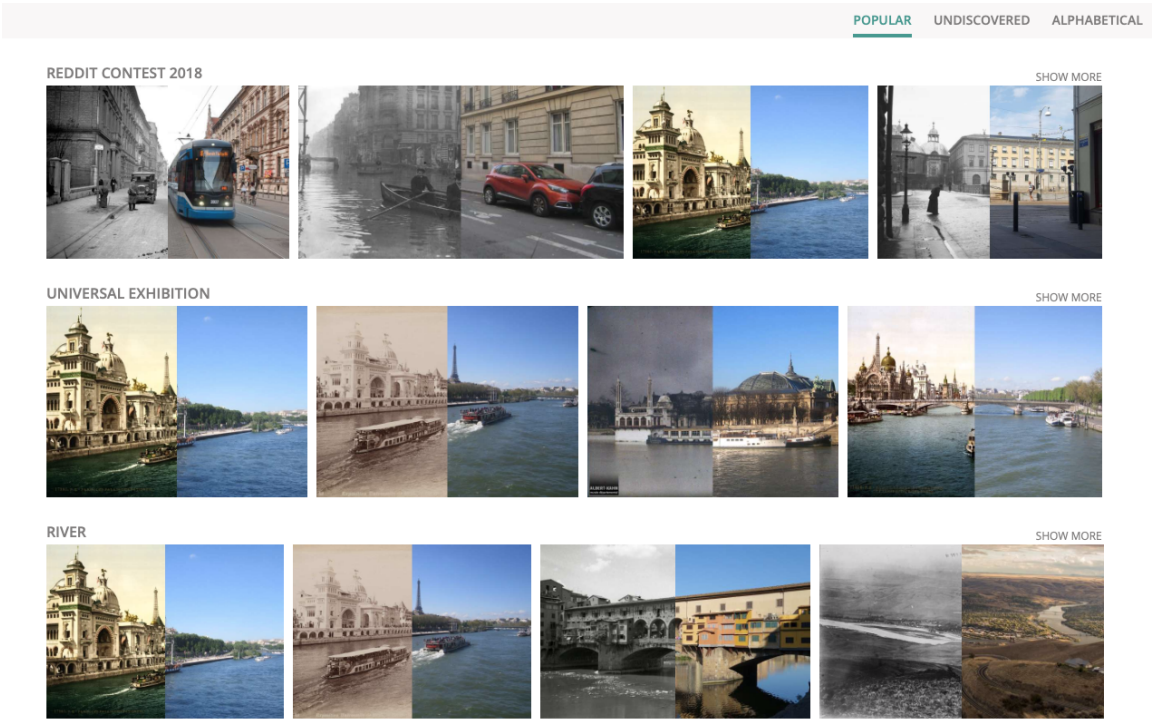


Figure 7.3: Illustration of browsing via popular image tags with previews of rephotographs belonging to each tag.

7.1.1 The Registration Process

As mentioned previously the portal allows photographers to register their images to achieve better alignment for image presentation. For this both images are presented next to each other, as shown in Figure 7.4 and the author is asked to mark corresponding points. To ease this process the image is magnified at the current position of the mouse cursor, while simultaneously the corresponding location in the other image is enlarged. Besides, correspondingly colored markers are assigned to all matching points selected by the user.

For alignment computation at least four points need to be marked, since a projective transformation between both images is computed, see also Figure 7.5. Yet, the user may select more points to improve results. Furthermore, it is necessary to select points at different depth of the image. This means, points in the foreground as well as the background should be selected. Instead, if all points belong to the same image plane, registration often results in large distortions towards image edges.

In the end, a homography matrix is computed based on all corresponding points marked by the author. This is used to transform the historic image, so that all four corresponding points overlap each other. In case more points were selected the overlap between all points is optimized. Finally, the resulting registration is presented to the user and he is able to publish his composition or repeat the alignment process to achieve a better result.

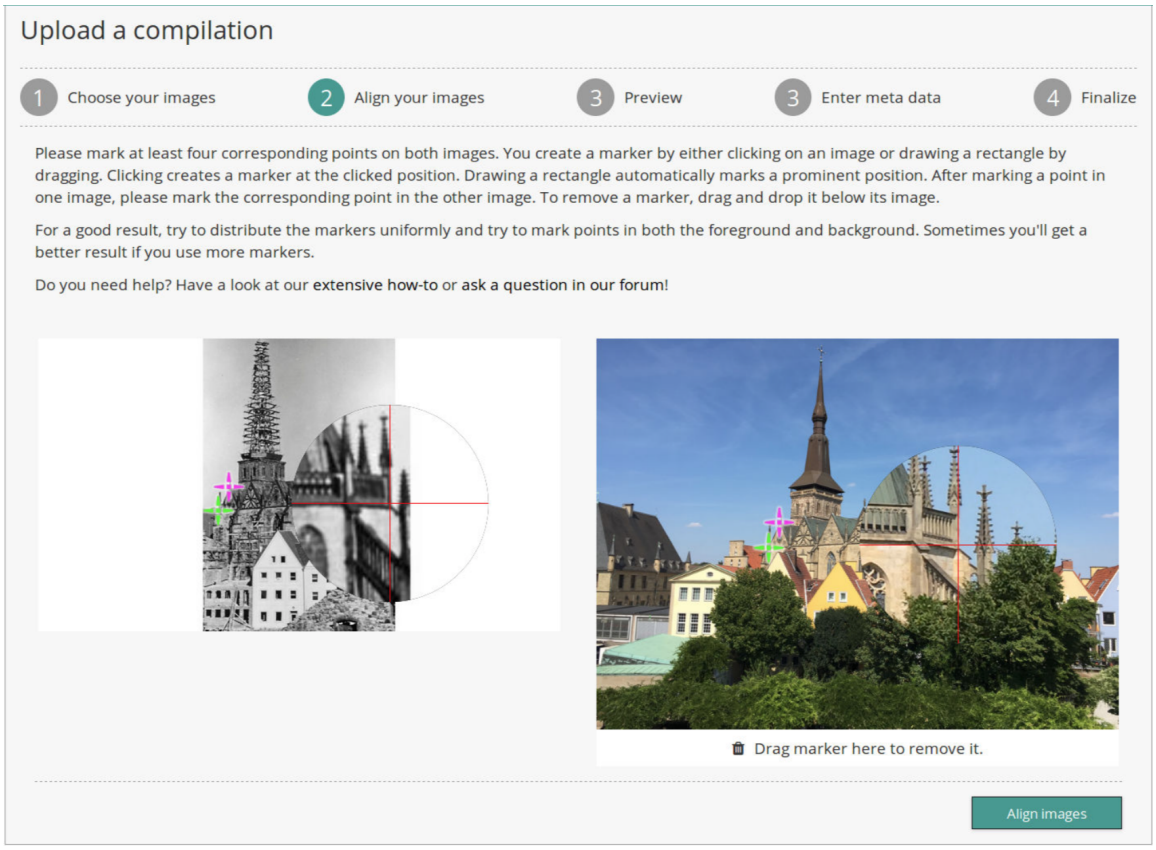


Figure 7.4: Illustration of the image registration process on re.photos. The image region at the mouse cursor as well as its corresponding region in the second image are magnified during marker setting. Image from [Schaffland et al., 2019].

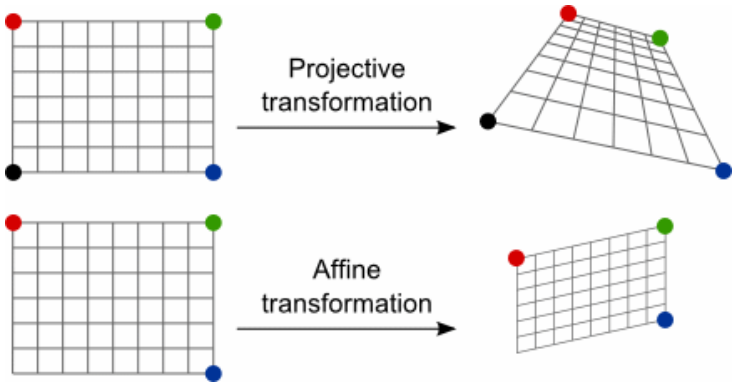


Figure 7.5: Illustration of a projective and an affine transformation[3]. While the affine transformation at the bottom preserves parallel lines and collinearity, a projective transformation preserves the straightness of lines, but their length and direction may be altered.

[3]reprint from `https://www.graphicsmill.com/docs/gm/affine-and-projective-transformations.htm` (accessed on January 7th, 2020)

Alternative Image Transformations

The decision to use a projective transformation for image registration is based on the following thoughts. A projective transformation is able to project planes to each other. Thus, if a rephotograph of a single plane is captured, we will always be able to achieve perfect alignment of this plane by post-processing it with a homography. All simpler transformations including rotation and scale change as well as affine transformations, which preserve parallelism and collinearity, are unable to achieve this.

In theory, a single building, captured from the distance, is close to a plane, so that it can successfully be registered by a projective transformation. Yet, in practice captured scenes commonly contain foreground as well as background objects, which both should be aligned in a perfect rephotography. This can not be achieved via a perspective transform due to the effects of parallaxe. Parallaxe refers to the angle that arises if an object is captured from two different views. Due to this it seems as if the object moved across the background between both images. An illustration of the effects of parallaxe during image registration is provided in Figure 7.6. Consequently, via preprocessing a perfect alignment can only be achieved for certain objects or parts of a rephotography.

Please note, image transformations accounting for parallaxe are much more complicated than the applied projective transformation. This is, since they need to decompose the image into different parts and register these individually. For this they require significantly more information on corresponding image regions than the four points currently marked by the user. Two alternative transformations we looked at in more detail are presented in the following.

Finally, the transformation is applied to the historic instead of the modern image, since this often suffers from low resolution and quality already and transforming the image does not create serious harm. Besides, after alignment both images are cropped, so that image pixels that are undefined after image translation are not visible in the published compilation.

Morphing

One alternative image transformation tested on several images from the re.photos portal is morphing using Delaunay triangulation. Morphing is commonly used in videos to crossfade faces. For our application a Delaunay triangulation is used to establish triangles between all points marked by the user. Afterwards, all marked points, now the corners of triangles, are aligned on top of each other and the area of each triangle is mapped to the area of its corresponding triangle via an affine transformation.

In theory, this individual transformation of each region allows to correct for all misalignment if enough points are marked. Yet, the independence of transformations is accompanied by discontinuities along triangle edges. This is illustrated in Figure 7.7. These discontinuities become particularly apparent if buildings are morphed since these contain many straight lines. After morphing these straight lines commonly bend along triangle edges. In practice the greater the transformation necessary the sharper the bend. In images featuring straight lines, such as photographs of urban scenes, this results in many artifacts, which lead to a surreal appearance of the entire scenery, that is not acceptable to the human eye.

Besides, to generate good results via morphing, that are able to account for parallaxe, a segmentation of the entire image into many small triangles is necessary. Consequently, the user is required to mark considerably more than four corresponding points, which is tedious and time consuming.

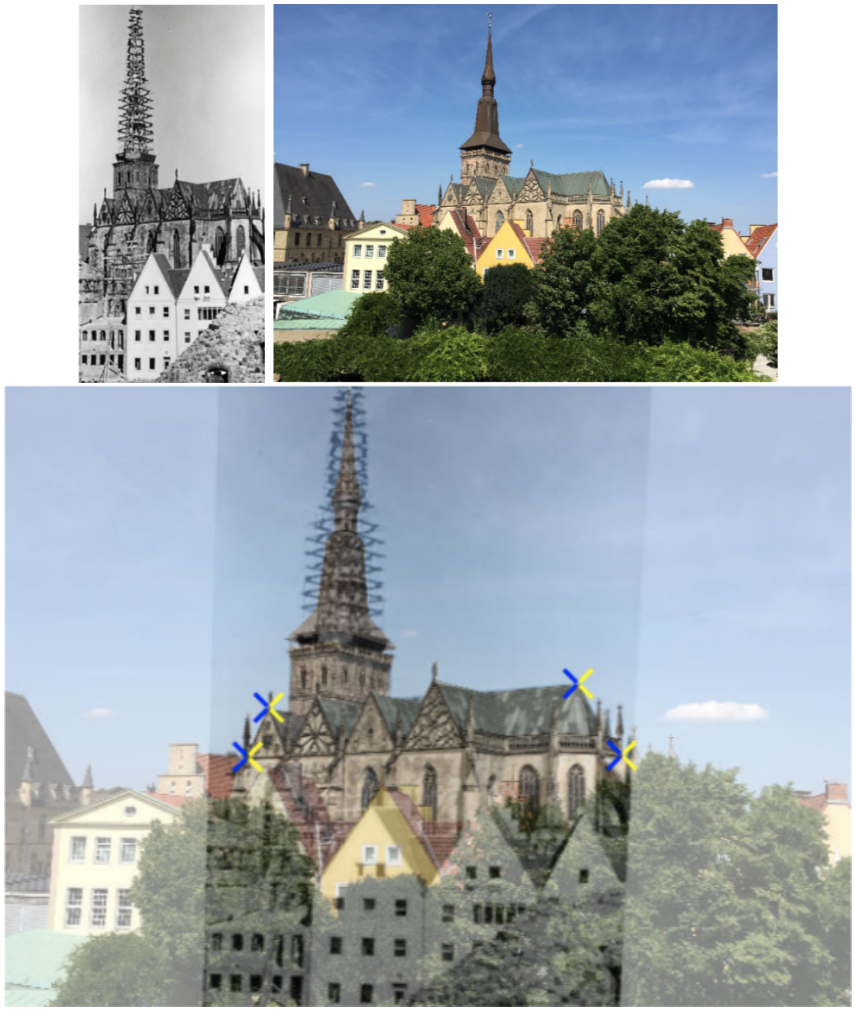


Figure 7.6: Illustration of the effects of parallaxe if a projective transformation is applied. At the top the original images are shown, while at the bottom their registration based on four markers is displayed. Since all markers were placed on the church this is well aligned, while the alignment of houses in the foreground suffers from the effects of parallaxe. Images from [Schaffland et al., 2019].

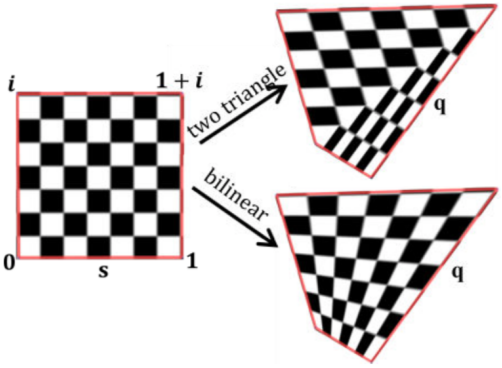


Figure 7.7: Comparison of two transformations of a square to a quad. At the top the morphing result is shown, if the square is separated into two triangles which are individually transformed. This leads to discontinuities along the triangle edges, which are avoided by a bilinear mapping, shown at the bottom. Image from [Chen and Gotsmann, 2016].

AAAP

Another approach to align rephotographs has been presented by Chen and Gotsmann [2016]. They apply an As-Affine-as-Possible (AAAP) mapping based on corresponding lines to reproduce the view of the historic photograph.

At first, the user is required to mark few corresponding lines in both images. Afterwards, a fine grid is placed across the image that is supposed to be transformed via AAAP. Now, the corresponding lines are aligned with each other, while all rectangles of the gird are deformed as little and as affine as possible. Examples are shown in Figure 7.8 and 7.9.

The individual transformation of each grid can compensate for many local deformations. Besides AAAP warping does not generate discontinuities at rectangle edges, as observed during morphing with Delaunay triangulation. However, while straight lines marked by the user are exactly aligned and remain straight, all other lines may be curved by the algorithm. This is illustrated in Figure 7.8, where the buildings below the spire are bent. Furthermore, if lines are marked in foreground and background objects and AAAP tries to correct for parallaxe, deformations and artifacts become visible, see Figure 7.9.

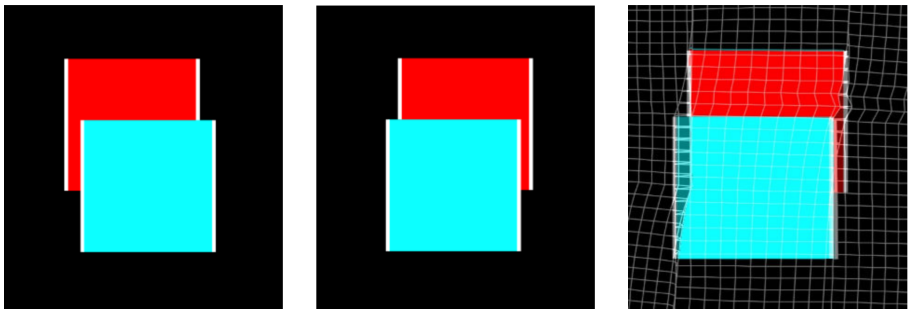
Another limitation of AAAP is that it is build to correct for small local transformations. Correcting for large global transformations such as perspective changes on the other hand, leads to strong deformations in image parts close to marked lines, see Figure 7.10. This is, since AAAP tries to reduce distortions for every rectangle of the entire grid. Thus, opposed to applying a projective transformation, displacement of image corners is avoided, while the inner areas of the image are subject to greater deformation.

(a) Original

(b) To be Transformed

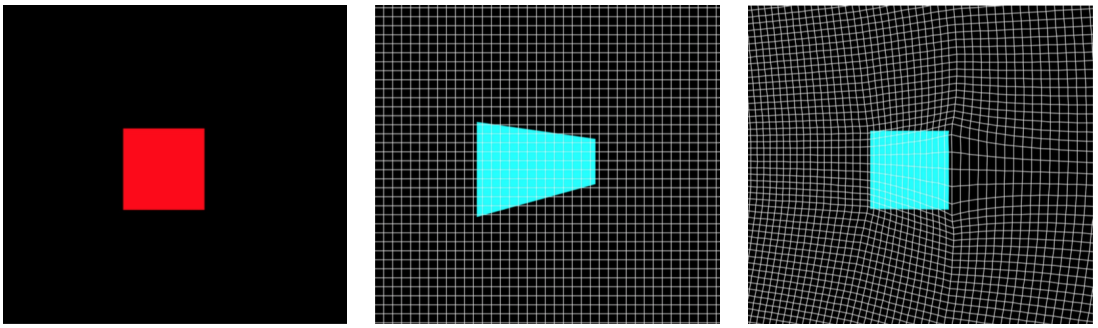
(c) Resulting Alignment

Figure 7.8: Illustration of alignment of an image pair via As-Affine-As-Possible (AAAP) warping.



(a) First View **(b)** Second View **(c)** AAAP Alignment

Figure 7.9: Illustration of deformations and artifacts after image alignment with As-Affine-As-Possible (AAAP) warping, caused by parallaxe. Images from [Schaffland et al., 2019]



(a) First View **(b)** Second View **(c)** AAAP Alignment

Figure 7.10: Illustration of deformations arising if As-Affine-As-Possible (AAAP) warping is used to correct for large changes of perspective. In this example the edges of the red square and the blue quad have been marked as corresponding lines. Thus, the object is aligned well, but the image area adjacent to the right edge of the object suffers from large deformations.

To avoid this, Schaffland proposed a two step approach. At first, the user is asked to mark four corresponding image points. Based on these a projective transformation is computed which corrects for large perspective changes between the images. Afterwards, the user identifies corresponding lines on the transformed image and AAAP is applied to correct for local distortions.

Summary

Utilizing morphing for image registration on re.photos is not an option. This is, since the user is required to mark many corresponding points, which is regularly reported to be tedious and time consuming. Furthermore, morphing with Delaunay triangulation results in image artifacts such as bends in straight lines which are unacceptable to the human viewer.

AAAP instead, requires the identification of corresponding lines, which can be marked quickly and few lines often suffice to cover the majority of the image. Additionally AAAP is more flexible than a single projective transformation and allows to correct for local distortion. However, the performance of AAAP suffers if an image does not contain enough corresponding straight lines, due to occlusion or the nature of the picture itself. Faces and rural landscapes without man made structures may contain no straight lines at all. Furthermore, since a two step approach is necessary to correct for large perspective changes, AAAP suffers from most issues users report during registration with a projective transform.

Registering two image pairs with a projective transformation is very easy since marking four corresponding points suffices to register both images. Yet, via a global transformation only certain parts of the image can be brought into perfect alignment, remember the problem of parallaxe. Furthermore, users report that if they mark more than four corresponding points to achieve better alignment, results are no longer comprehensible. Consequently, if the points initially marked by the user do not result in good alignment, users do not know how to improve the registration.

Especially, the latter challenge remains, even if AAAP is applied after an initial projective transformation. Thus, at first the current registration process used on re.photos, which aligns images via a projective transformation, needs to be improved and become more traceable for the user. After this, additional AAAP application may be considered.

7.1.2 Advancements to the Registration Process

Users report two main drawbacks of the registration process initially employed on the re.photos portal, involving completely manual selection of corresponding points as well as alignment via a projective transformation. At first, the exact selection of corresponding points is very tedious, despite the use of a magnifying glass. Second, the final outcome of the registration process appears not comprehensible to the user if more than four corresponding points are marked.

The research outlined in Chapter 4 and 6 showed, that currently a complete automation of similarity detection during the registration process is too error-prone to be utilized in a practical application. Thus, as an intermediate step, we try to ease the manual registration process of the re.photos portal, by accelerating corresponding point selection. Furthermore, interactive registration process is presented, that is more traceable for the user.

Automatic Corner Detection

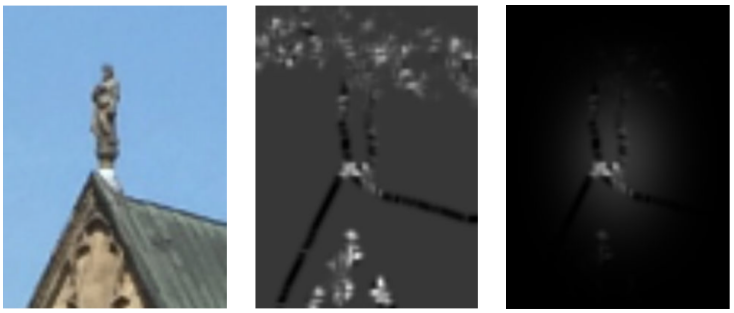
In general, corresponding points selected during the registration process are corners of buildings, building parts (e.g. windows) or other objects in the image. In the literature there are several approaches to automatically detect image corners. A popular method is the Harris Corner Detector [Harris and Stephens, 1988], which we use to accelerate point selection.

As an alternative to directly selecting a point, we allow the user to span a rectangle around the corner he intends to select. Now, we apply Harris Corner Detector to identify all pixels belonging to corners in the marked image area. Afterwards, we weigh all identified corners by their distance to the image center with a Gaussian distribution, see also Figure 7.11. The weighting is necessary, since regularly more than a single corner is detected inside the rectangle. Yet, it is assumed, that the corner the user intended to select is located in the rectangles center.

This extension to the registration process is currently used by the re.photos portal and speeds up point selection significantly. Before the user was required to first place a point approximately and then move it to the exact location with the help of the magnifying glass. Now the quick selection of a rectangular area via drag and drop suffices to mark a point. Only in rare cases an additional readjusting of the point with the help of the magnifying glass is necessary. Yet, even if this is the case the procedure does only require two steps, as previous direct point selection.

Automatic Corresponding Point Detection

However, the presented procedure still requires the user to mark corresponding points in both images individually. Additionally, an approach was tested, that automatically detects a corresponding point in the second image after the user marked one in the first image. For this we



(a) (b) (c)

Figure 7.11: Illustration of automatic corner detection. (a) Area marked by the user. (b) Result of Harris Corner Detector. (c) Selection after weighting all corners by their distance to the image center. Images from [Schaffland et al., 2019].

assume, that the corresponding point in the second image is located close to the pixel position of the point in the first image.

In detail a large rectangular area from the second image around the pixel position of the selected point in the first image is taken into account. Inside this the algorithm again searches for potential corners with Harris Corner Detector. Second, template matching [Brunelli, 2009] is used to compare the rectangular region around the marked point in the first image with rectangular regions around all potential corners detected in the second image. Finally, the best match is selected as the corresponding corner.

This procedure has several drawbacks. First of all, it is not invariant to rotation, scale or perspective change. Second, both images have to be aligned to a certain degree beforehand. Otherwise, the corresponding corner searched for is not part of the rectangle selected from the second image and can not be found. This is an issue especially if both images feature different image width and height. Indeed, it is not clear, were to position the rectangle in the second image if such has an upright format, while the first image is in landscape format. Third, if the selected point belongs to a repeating pattern such as a front of windows it is extremely difficult to identify its correct corresponding corner. Unfortunately, especially in urban rephotograph repeated patterns are very common.

Consequently, to utilize this approach on the re.photos portal, it is necessary to split the registration process into two steps. At first, up to four points have to be individually marked by the user to establish a rough alignment of both images. Only after this more points can be selected via automatic correspondence detection. This is rather complex, especially keeping in mind that many users only mark a total of four points. Thus, they would never use the automatic correspondence detection. Yet, if only two points are selected in the first step, correcting for translation, rotation and scale changes but not perspective change, automatic correspondence selection is likely to fail for all images pairs with greater view changes.

Another issue is, that a large fraction of users tend to omit the readjustment of automatically marked points. Thus, if automatic corresponding point selection identifies the wrong point, which as mentioned is a common problem, users tend to keep the automatic selection. In this case the final alignment computed is rather bad and users become easily frustrated. On the other hand, more eager users who regularly readjust automatically selected correspondences, may get the impression that automatic point selection does not help them at all.

Thus, we currently refrain from utilizing this form of automatic corresponding point selection during the registration process.

Automatic (Corresponding) Edge Detection

In the context of exploring As-Affine-As-Possible (AAAP) warping, we additionally developed a method to ease the selection of corresponding lines. With this the user is only required to draw a line close to the edge he wants to select and we automatically readjust this line to the closest image edge.

The method we apply for this is very similar to automatic point selection. At first, a rectangular region is selected around the line marked by the user. In this region we detect potential edge pixels with the Canny Edge Detector [Canny, 1986]. Afterwards, we apply a filter to select all edges featuring a gradient similar to that of the marked line. Now the Hough transformation is used to extract lines based on the remaining edge pixels [Duda and Hart, 1972]. Finally, all detected lines are compared to the location and length of the line marked by the user and the best fit is selected.

First tests showed that this procedure is rather robust against false edge selection. This is, since edges of a specific direction are more unique than corners, especially if their length is taken into account. Besides, if two potential edges have been detected it can be assumed that the one closer to the marking of the user is the desired one. Only if a thin pillar like object is marked, it is sometimes difficult to decide which of its two equivalently long edges the user intended to select. Thus, in case AAAP is used for registration, automatic readjusting of edges can accelerate corresponding line selection.

Furthermore, along the lines of automatic corresponding point detection, we developed a similar method for automatic corresponding edge detection. This finds potential edges in the second image around the location of the first image and applies template matching to determine the best fit. Yet, as automatic corresponding point selection, this method requires images to be roughly aligned beforehand. However, since the application of AAAP in general requires well aligned images resulting in a two step approach, this is no longer an issue. Besides, initial results show that corresponding edge detection is more robust than corresponding point detection, due to the greater uniqueness of lines. Nonetheless, the approach should be evaluated on a broader basis and tested by several users before integration into any practical application.

Interactive Registration

Besides, simplifying the registration process via faster point selection another challenge is to make the entire process more transparent to the user. For this purpose Schaffland et al. [2019] developed an interactive registration method. This immediately visualizes the effect the previously selected point correspondence has on the alignment of both image pairs. Furthermore, after selecting four corresponding point pairs the user is able to readjust these point selections and again immediately view the results on alignment computation. Thus, the user is able to retrace the registration process and may correct for undesired results immediately.

In detail the entire registration process is split into four steps. (1) At first, a single corresponding point pair is selected by the user. Based on this one image is translated so that both selected points overlap. (2) Next, the user selects a second corresponding point pair. Now a transformation scaling and rotating one of the images is computed so that the current as well as the point pair from the previous step are aligned. (3) After this, two more point pairs need to be selected by the user to compute a projective transformation that perfectly aligns all four point pairs. (4) Finally, the user is able to view the registered images, as shown in Figure 7.6, and realign individual corresponding point pairs. This is achieved as the user selects one point pair and drags one of its points to the desired location, while the image registration is continuously updated.

The described interactive registration process allows the user to select a maximum of four corresponding point pairs, while he is able to terminate the registration process at an earlier stage if sufficient alignment has already been reached. On the other hand, interactive registration does not allow more than four point pairs to be selected, since with more points the projective transformation computed is no longer definite, but the result of an optimization procedure. As a consequence, results are no longer predictable and hardly traceable for the user. Yet, so far this interactive registration process has only been proposed and still needs to be evaluated in a user study before integration into the re.photos portal.

Outlook

This section presents further ideas to expand the registration process, which need to be implemented and evaluated in future work.

Manual Cropping: The first idea is to allow the user to manually crop his rephotographic image composition after alignment. Currently, only image regions undefined after image registration are automatically removed. Instead, if the user is able to crop images he may also remove image frames as they are often present in postcards. Furthermore, he might decide to present only those image parts, which are well aligned in the final compilation and cut out some background or foreground objects as well as misaligned image edges.

More flexible Presentation: In the context of automatic image cropping also the opposite phenomenon arises. This is, that great perspective changes are necessary to align two rephotographic images and as a result large image parts are cropped, even though this is not desired by the user. To get around this we could allow to present compilations of images featuring different sizes on the re.photos portal. During the current presentation this would result in at least one image being surrounded by undefined (black or white) image regions. To avoid this undefined image regions could be replaced by the defined pixels of the second image. This would lead to a more *Timera* like view of one of the images of the compilation, see Figure 7.12. In combination with individual cropping of both images this provides more artistic license to the user.

Figure 7.12: Rephotograph from *Timera*[4]: Kure Beach, United States by Jo Rockstar.

[4] `http://www.timera.com/t/68ugaEmo` (accessed on April 30th, 2017)

Optional AAAP application as step 5: In case the interactive registration process is integrated into the re.photos portal such can be expanded by an optional 5th step. During this As-Affine-As-Possible warping (AAAP), with automatic edge detection next to a user defined region, can be used to improve the alignment of individual image segments. As in the previous steps of the interactive registration process image deformations resulting from marked edges can be made visible immediately. Besides, in this context most disadvantages of AAAP mentioned previously are evened out. Since in step 4 a projective transformation has already been applied to the image pair to correct for global perspective changes, the two step procedure AAAP requires to correct for larger perspective changes is utilized. Furthermore, if not enough lines are present in the image, or if it appears to time consuming for the user, AAAP application can simply be omitted. Thus AAAP application can be an additional tool for dedicated users to improve their image compilations even further.

Corresponding point detection utilizing feature descriptors: During the interactive registration process the previously described approach to automatic corresponding point detection can be utilized in step 3, at which a rough alignment of both images has been achieved via translation, rotation and scaling. However, in the future an advanced automatic corresponding point detection approach based on the findings of Chapter 4 and 6 of this thesis can be developed. In detail, the entire corresponding point selection process would still be semi-automatic, since the user is required to select a point in the first image. Yet, a greater part of the second image, possibly the entire image, may be searched for a corresponding point, if instead of template matching feature descriptors are used to select the best matching corner. These feature descriptors are more robust against changes in appearance including different lighting and occlusion, as well as against scale and perspective changes. Depending on the computational power available LATCH descriptor (Chapter 4) or CNN features from conv3 of VGG-16 (Chapter 6) may be used. Utilizing feature descriptors instead of template matching can lead to major improvements of automatic corresponding point selection and may allow its exploitation at steps 1 and 2 of the interactive registration process as well.

7.2 Mobile Applications for Assisted Rephotography

The re.photos portal allows users to align their rephotographs more accurately via post-processing. However, as outlined post-processing is not able to correct for all misalignment of an image pair. Parallaxe is a common problem, which leads to a bad registration of some parts of the final rephotographic compilation. In general, the closer the viewpoint of the original and its rephotograph, the better the output of post-processing. Consequently, it is necessary to support the users during their search for the exact viewpoint of an original photograph.

7.2.1 A General Mobile Application

Since rephotographs are usually taken in the field, the goal is to develop a mobile application for smartphones and tablets, that guides the user to the original viewpoint of the photograph he intends to retake. So far several prototypes have been developed [Diederichsen, 2015; Köhler, 2014; Steinkamp, 2016; von Behren, 2017]. They allow the user to load the image they intend to recapture from their gallery. Now the edges of this image are detected and overlaid on top of the current camera view, see Figure 7.13. This visualization eases the process of relocating the exact view of the original image. Finally, after capturing the rephotograph, the user is able to improve the registration of both images via marking corresponding points, as in the registration process on the re.photos portal.

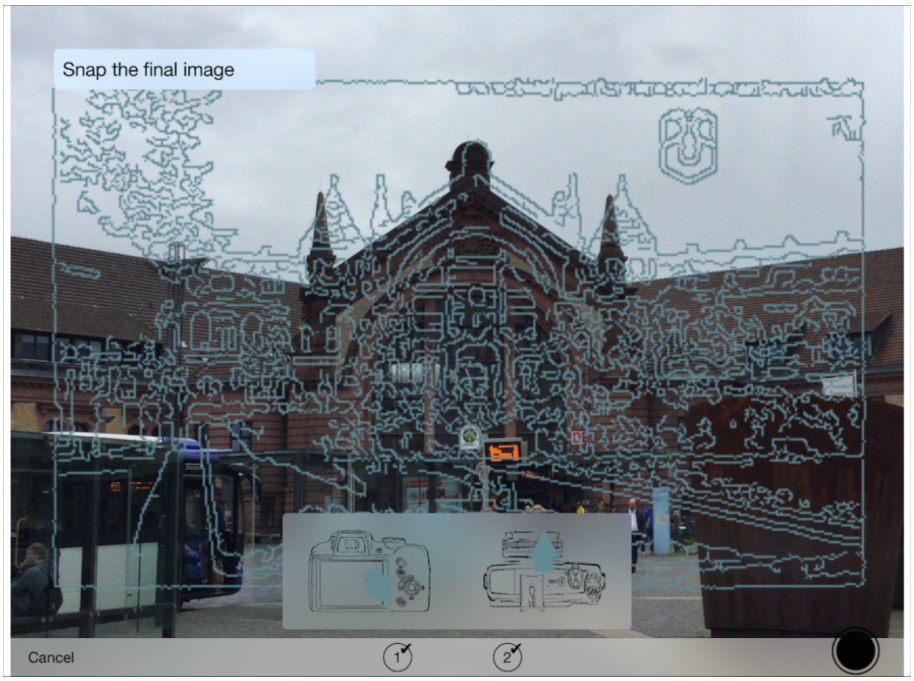


Figure 7.13: Illustration of the capturing view of the mobile applications developed by Diederichsen [2015]

.

Köhler [2014] developed the first prototype of such a mobile application, independent of the re.photos portal. This showed that an edge overlay improves recapturing of the original viewpoint, yet often it is still very difficult for the user to determine the correct direction of movement. Thus, even with the mobile application, the recapturing procedure still involves trying out several movements into different directions, before successful approximation of the original view. Besides, we realized that manual marking of corresponding points for post-processing is very fiddly on the small displays of tablets and smartphones operated by finger touch. In fact precise marking of points is only possible if a touch pen is used.

Steinkamp [2016] developed a successor application connected to the re.photos portal. This allows users to select original images from the online archive of re.photos and to upload generated rephotographic pairs to the portal. Upon this users can decide, whether they intend to directly publish their compilation or prefer to further edit it on the computer. Thus, it is easy to use the mobile application for capturing a rephotograph and proceed with the post processing of both images on the computer, were marking of corresponding points is possible with the accuracy of a mouse cursor.

Diederichsen [2015] on the other hand focused on enhancing user guidance. On the basis of Bae et al. [2010] he developed a mobile application, which shows the user the direction of movement required to recover the view of the original photograph. His prototype revealed several open challenges requiring a solution before such an application, providing direct guidance, can be released to users without further instructions. A detailed description of his approach and the challenges faced is presented in the following.

7.2.2 User Guidance

To provide the user with information on the required direction of movement he has to take two additional shots of the scene he intends to recapture. The first shot needs to be taken from a far perspective approximately 20 degrees away from the original image. The second shot is supposed to be taken from a view as close to the original as possible. These two shots featuring a wide baseline allow a reliable reconstruction of the scene via classic feature matching. Afterwards, corresponding points between the original reference image and the second shot are identified. This allows the estimation of relative movement between the first shot and the reference image.

To avoid motion degeneracy, in the following the current camera frame is always compared to the first shot (20 degrees away from the original). Now, with the location of the current frame and the reference image relative to the first shot, the relative movement between the current frame and the reference image can be computed. The required movement is displayed to the user via two arrows, one visualizing the required translation in the XY-plane (right, left, up and down), while the other shows the movement required in the Z-plane (forward and backward). This visualization has been adapted from Bae et al. [2010] and is illustrated at the bottom of Figure 7.13. The whole pipeline used to generate the required motion is shown in Figure 7.14.

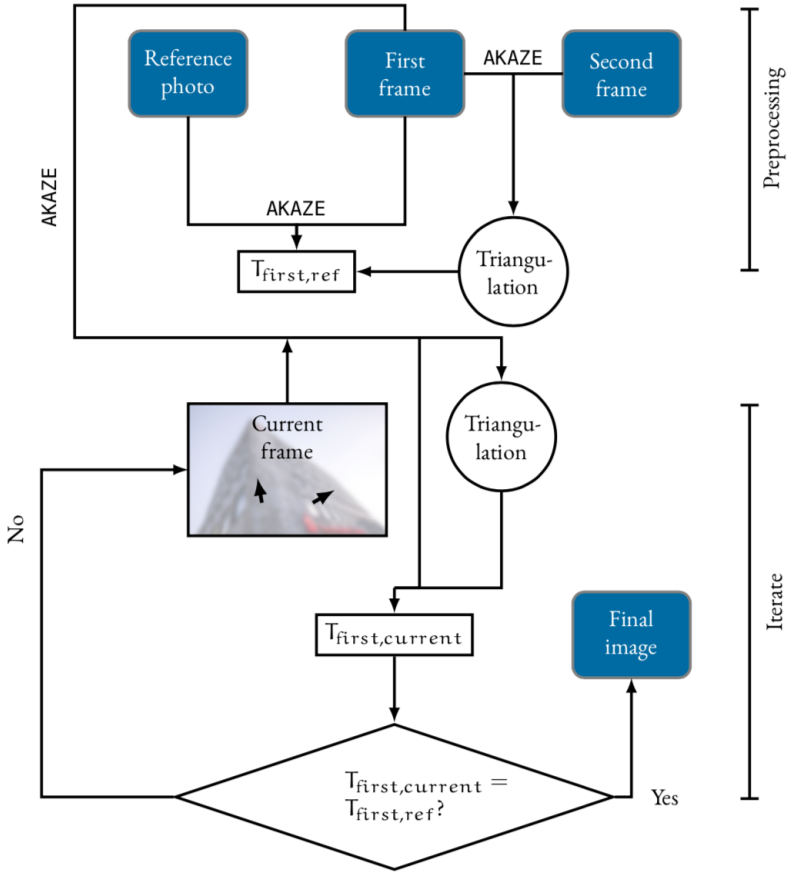


Figure 7.14: Illustration of the computational rephotography pipeline applied in the mobile application of Diederichsen [2015]

.

7.2.3 Challenges

The evaluation of this approach revealed the following challenges for future work:

1. Scale and rotation are successfully estimated, as long as the photographer does not move entirely along the optical axis (forward and backward only). However, the most important component, the translation specifying the required direction of movement, cannot be recovered at all and is off by 80 degrees on average. This is a result of the equation used to compute the required translation:

$$T_{ref,current} = -R_{ref,first} * R^{T}_{current,first} * T_{current,first} + T_{ref,first} \tag{7.1}$$

   Often the user mostly moves into a single direction between the first shot and the reference image. In this case both summands of equation 7.1 feature the same dominant orientation, which will be zeroed out after subtraction. The resulting vector is normalized to unit length and scaled with the previously determined factor. Thus, small differences in the orientation of both vectors are enhanced and the final translation vector computed points into a completely wrong direction.
   A possible solution to avoid this is to compute all position estimates relative to a 3D model of the scene. This can be established based on the first and second shot and if necessary be refined with the input from new camera frames. Via 3D to 2D point correspondences $T_{current,first}$ and $T_{ref,first}$ can be directly obtained with a corresponding scaling. Thus, the enhancement of non dominant directions of movement is avoided. This approach needs to be evaluated in future work, including strategies to handle the limited availability of computational power on mobile devices.

2. Another issue is related to automatic feature detection. Automatic feature detection and matching in each camera frame results in different sets of features for each current frame. As a result, in the approach of Diederichsen [2015] and Bae et al. [2010] scale estimation may vary across frames.
   One solution can be to perform feature tracking across all camera frames instead of feature detection from scratch. This has already been tested by Justin Shenk and works reliable as long as features do not disappear from the camera's sight. A positive side effect is that feature tracking is less computational intensive than repeated feature detection and matching. However, since the features detected in the first and second shot are used for tracking until recapture of the original image, it needs to be made sure that this set of features is well suited for pose estimation.
   Alternatively, as mentioned previously, a complete 3D model of the scene can be established and updated upon new incoming camera frames. As a result, scales are estimated with reference to this 3D model and should stay constant across the recapturing process. Yet, in comparison to the original approach, computation and refinement of an entire 3D model of the scene is computationally expensive.

3. If a scene mainly consists of a single dominant building and only very few foreground elements, as shown in Figure 7.15, pose estimation and 3D model construction suffer from degeneration. This is, since during optimization the few existing foreground features are regularly classified as outliers to obtain a solution better fitting the majority of background features. However, if all features of the background object lie more or less on a single plain, adequate model construction and pose estimation fails.

Figure 7.15: Example of a scene with a dominant object in the background and only few elements in the foreground, for which relative pose estimation based on automatic feature detection regularly fails. Images from Diederichsen [2015]

.

This might be avoided by extending the effective match distribution measure, developed in Chapter 5, to 3D and ranking the possible 3D models not only by the number of inliers but also their distribution across image depth. However, the details of this approach as well as its ability to solve the described issue need to be developed and evaluated in future work.

4. Against the results of the user study reported by Bae et al. [2010], we experienced, that the used visualization with two cameras and two arrows as shown in Figure 7.13, is not intuitive. Thus, follow up work should explore more options to visualize the required movement to the user and evaluate all proposed methods against in a broad user study.
   One idea is to project a rectangle, replicating the camera location at which the rephotograph needs to be taken, into the current camera view, as shown in Figure 7.16. However, this requires a very accurate reconstruction of the original images location in 3D as well as further ideas how to represent the required movement if the target rectangle is not visible in the current camera view.
   Another approach would be to display only a single arrow in 3D to the user at all times, as shown in Figure 7.17. This can either represent the required movement in the XZ-plane (right, left, forward and backward), or in case the height of the camera needs to be adjusted only show the required movement in the Y-plane (up and down).
   Furthermore, the full edge overlay generated from the edges of the original image, may be supplemented by single points marking corresponding features between the original image and the current camera view. In the beginning, these features are picked to align the original often historic camera to the second shot. Afterwards, these only need to be tracked across camera frames. This way the user can easily see, which points still need to be brought into alignment, before capturing the rephotograph.

Shi et al. [2018]

Recently, Shi et al. [2018] developed a mobile application for rephotography, that provides the user with nearly real-time navigation information. Originally, their application aimed at monitoring minute changes of ancient murals, which are usually planar. So their navigation clues are

Figure 7.16: Illustration of user navigation via a rectangle, which replicates the position the original image was taken from.

Figure 7.17: Illustration of user navigation via a single arrow. In the left image possible navigation instructions in the XZ-plane (right, left, forward and backward) are displayed. Superimposition with the yellow arrow suggests forward movement, while superimposition by the green arrow suggest to move to the front right. During the application only one arrow would be visible at a time. The right image on the other hand displays movement required in the y-plane (up and down). The circle of arrows shows the user that he should stay at his position, but needs to raise his arms and move the camera upwards.

based on homography estimations between the reference image and the current camera frame. Across camera frames they switch between a reliable ORB feature and RANSAC based matching and a flow-based fast matching, which tracks features across frames. This efficiently decreases computational complexity, which is an issue on mobile devices. Furthermore, they introduce a robust keyframe decision strategy and compute $H_{current,reference}$ directly based on the feature matching set $P_{current,reference}$ to increase accuracy and robustness against illumination changes. In the final evaluation they show, that for planar or near planar scenes their approach outperforms linear-blending and homography-based rephotography [Feng et al., 2015]. Besides, they illustrate that their method can work well on 3D scenes.

However, it is obvious that in 3D scenes the applicability of Shi et al.'s [2018] approach is limited. Since navigation clues are only based on homography estimation the application is not able to guide a user to the exact view point of a 3D scene and correct for parallaxe. Furthermore, their approach is robust against illumination changes but does not account for high occlusion or large structural changes between the reference image and the current scene appearance. Thus,

their current application is unable to handle the kind of historic images we are dealing with in the nyc-grid dataset as well as on the re.photos portal.

Yet, the work of Shi et al. [2018] provides several ideas to advance our mobile application, currently based on the work of Bae et al. [2010]. As mentioned, future work should build a 3D point model of the current scene, which is regularly refined as the user approaches the target position. Thus, the relative position between the current and target position can be directly computed from the 3D model. This is computationally expensive and cannot be done for every incoming camera frame in real time. Hence, as envisioned previously and proposed in Shi et al. [2018], the mobile application needs to switch between two modes for generating navigation clues, namely fast feature tracking and more reliable 3D model refinement. Furthermore, the robust keyframe decision strategy can be adapted to select a suitable subset of frames for accurate 3D model construction.

Finally, while the approach of Bae et al. [2010] and current implementations fail to handle planar or near planar scenes, Shi et al. [2018] handle this type of scenes very well. However, in historic rephotography near planar scenes are common as well, for instance if the front of a single building is captured from far away. As a consequence, an ideal mobile application for rephotography would implement both approaches, differentiate between near planar and 3D scenes and use the appropriate algorithm to guide its user.

7.2.4 Further Outlook

After several ideas regarding user guidance have already been presented on the previous pages, this section focuses on possible enhancements for the remaining aspects of the mobile application. Furthermore, an idea for a mobile application is proposed, which provides a new way to consume rephotographs already present on the re.photos portal.

At first, follow up work needs to integrate the mechanism of automatic corner detection already used by the re.photos portal and presented in Section 7.1.2. This will aid users in accurately marking corresponding points by finger touch on the small displays of mobile devices. Additionally, if proven to be a success on the re.photos portal, the interactive registration process also needs to be integrated into its accompanying mobile application.

Another idea is to allow the user to select several historic images from the online archive of the re.photos portal. Provided with these the mobile application plans an entire tour, which allows the user to move from scene to scene and take several rephotographs of close by locations. In an ideal scenario this tour should even take into account information about the position of the sun and the shooting direction of each image, so that the user is not guided to a certain scene he needs to recapture against sunlight.

Rendering of Inaccessible Rephotographs

A further challenge for rephotographers is that sometimes the original view of a historic photograph is no longer accessible. This may be the case if a new building was constructed at the capturing spot or now a main street runs across it. Less frequently capturing spots also disappear, such as a balcony of a building which no longer exists. Currently, in these cases the rephotographer either needs to resign from creating this particular composition or take a picture from as close to the original view as possible and hope to be able to correct most artifacts via post-processing. On the other hand, with computational support and methods such as SfM and image rendering artificial views of scenes can be generated as Lee et al. [2011] showed. However,

depending on the images available for reconstruction, the quality of a rendered rephotograph varies. Thus, a future goal for the mobile application could be the following.

During viewpoint recovery the user is able to inform the mobile application that the original view point is not accessible. In this case the application switches to a new mode which aims at collecting a qualitative set of images for rendering the original view. Thus, the app does no longer try to guide the user to the original viewpoint, but lets him circle around the original view point and capture several images from surrounding views. Occasionally these images are used to refine the 3D model of the scene and the original image view is rendered based on all captured images. In case parts of the rendered image are still undefined the mobile application navigates the user to further locations to take additional shots filling these gaps. Finally, an artificial rephotograph with as few artifacts as possible should be created.

Augmented Reality City Tours

Another idea is to create a second mobile application targeted at consumers instead of rephotographers. This application should provide users with more interactive ways to experience the compilations displayed on the re.photos portal. Initially, a user may select his particular interests, such as popular sights, churches, a certain epoch or more natural landscapes. In the following he chooses his current location or a city he intends to visit as well as a desired time frame. Afterwards, the application designs an individual tour tailored to the users interests at which he visits several scenes recaptured on the re.photos portal. Now, as soon as the user reaches a certain location and points his mobile device at a certain building or scene, the original view of this scene is displayed. Thus, the changes a city or certain location experienced across time become even more tangible to viewers.

7.3 Summary

Previous chapters showed that acquiring and post-processing rephotographs is far from being fully automatized. However, several possibilities to partially automate and thus ease the process of capturing rephotographs exist. These may not only benefit ecologists and other scientists, who utilize rephotography to monitor changes in the environment, but also the general public.

In this section the re.photos portal was introduced, which allows users to upload, register and present their own rephotographs. Furthermore, several mobile applications were presented, which aid users in capturing rephotographs, by supporting the recovery of the original vantage point of the reference image. Besides, open challenges as well as possible solutions, regarding the registration process on the portal as well as the developed mobile applications, were listed.

Overall, as the use of the re.photos portal shows, several people recapture scenes as a hobby and even more are interested in viewing rephotographic compilations. Consequently, the portal as well as corresponding mobile applications should be developed further, to make it attractive for even more users to create, share and consume rephotography.

Chapter 8

Summary and Conclusion

This thesis tackled the challenge of registering modern to historic images in the context of urban rephotography. One of its major goals was to automatically identify similarities in historic and modern scenes, which have been exposed to medium to tremendous changes across the years. This should allow efficient post-processing of rephotographs as well as assistance in viewpoint recovery during image capture.

In related works classic features such as SIFT [Lowe, 2004] and SURF [Bay et al., 2006] were applied to match modern and historic images [Ali and Whitehead, 2014; Bae et al., 2010; Gat et al., 2011; Hauagge and Snavely, 2012; Schindler and Dellaert, 2012]. However, no consistent opinion on the suitability of classic features for this task existed. This is since specific feature matching pipelines [Ali and Whitehead, 2014; Schindler and Dellaert, 2012] or single parts of them [Gat et al., 2011] were evaluated on different datasets and the effects of combining different features and filters were not assessed.

Thus, at first this thesis presented a detailed evaluation of the performance of diverse detector and descriptor combinations, as well as different match filtering approaches for registering rephotographs. The study was conducted on a new dataset containing 52 rephotographs of Manhattan spanning time periods from 3 up to 100 years. It revealed that keypoints detected by local feature detectors are not stable across long time periods and a dense sampling of keypoints is preferable, while the feature descriptors RootSIFT [Arandjelović and Zisserman, 2012] and LATCH [Levi and Hassner, 2016] showed good performance. Besides, it was shown that common match filters based on descriptor distance are not suitable in the context of historic and modern image matching and instead filters based on geometry need to be applied. Furthermore, all model estimation approaches, including RANSAC [Fischler and Bolles, 1981] and ORSA [Moisan and Stival, 2004], were not able to reliably detect outliers. Consequently, applying good filters previous to model estimation is very important. Finally, major structural changes of a scene were identified as the biggest challenge for rephotographic image matching, which may also occur after short time spans. All these findings are not only relevant in the context of rephotography, but also apply to other challenging image pairs, as the successful application of the presented approaches to another dataset [Hauagge and Snavely, 2012] showed. However, even considering the performance of the most successful method, there is still room for improvement.

During the assessment of this first evaluation, research in the related field of place recognition prospered, which focused on season and illumination, but not long time changes [Lowry et al., 2016]. Studies showed that features extracted from Convolutional Neural Networks (CNNs), pre-trained on large datasets, outperform classic handcrafted features in challenging conditions such as textureless indoor scenes [Taira et al., 2018] or day time and season

changes [Sattler et al., 2018]. Additionally, semantic scene understanding was exploited to enhance location recognition [Garg et al., 2018; Schönberger et al., 2018; Toft et al., 2018]. Hence, this thesis presented a second evaluation, which assessed the suitability of pre-trained CNNs for aligning historic and modern image pairs, and compared different approaches utilizing semantic annotations to enhance matching performance. The results prove the general applicability of features extracted from mid-level layers of CNNs, originally trained for location recognition, for aligning historic and modern images. Besides, semantic information is able to enhance the alignment of rephotographs, especially in terms of accuracy, but improvements are not remarkable. Finally, the construction of a new training set is proposed, since most performance shortcomings can be explained by the composition of the dataset the applied CNN was pre-trained with.

In both evaluations, a new method to assess the quality of an automatically computed perspective transformation was presented. Furthermore, an advanced method to measure match distribution across an image was developed.

At last, an online portal which allows users to upload, register and present their own rephotographs was introduced. Currently this portal contains more than 2000 image pairs which may contribute to a larger dataset in the future. In addition, the first version of a mobile application that supports recovering the original viewpoint of an image was presented. Moreover, open challenges and possible future solutions regarding the registration process on the portal as well as the developed mobile application were discussed.

In summary, this thesis presented fundamental investigations in computational rephotography, by assessing the applicability of state-of-the-art approaches from related disciplines in the context of historic to modern image registration. In the future, further development of the presented approaches can simplify the process of taking rephotographs. Besides, the investigated methods may contribute to automate tasks beyond rephotography, including the sorting of image archives or navigation in disaster zones, which experienced severe structural damages.

Bibliography

S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski. Building Rome in a Day. *Communications of the ACM*, 54(10):105–112, 2011.

M. Agrawal, K. Konolige, and M. R. Blas. CenSurE: Center Surround Extremas for Realtime Feature Detection and Matching. *European Computer Vision Conference (ECCV)*, pages 102–115, 2008.

A. Alahi, R. Ortiz, and P. Vandergheynst. FREAK: Fast Retina Keypoint. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 510–517, 2012.

P. Alcantarilla, A. Bartoli, and A. Davison. KAZE Features. *European Conference on Computer Vision (ECCV)*, pages 214–227, 2012.

P. F. Alcantarilla and T. Solutions. Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces. *Transactions on Pattern Analysis and Machine Intelligence*, 34(7):1281–1298, 2011.

H. K. Ali. *"Timescape Image Panorama Registration Techniques"*. PhD thesis, Carleton University, 2016.

H. K. Ali and A. Whitehead. Feature Matching for Aligning Historical and Modern Images. *International Journal of Computers and Their Applications*, 21(3):188–201, 2014.

H. K. Ali and A. Whitehead. Registration of Modern and Historic Imagery for Timescape Creation. *Conference on Computer and Robot Vision (CRV)*, pages 124–131, 2016.

A. Angeli, D. Filliat, S. Doncieux, and J.-A. Meyer. A Fast and Incremental Method for Loop-Closure Detection using Bags of Visual Words. *IEEE Transactions on Robotics*, pages 1027–1037, 2008.

A. Anoosheh, T. Sattler, R. Timofte, M. Pollefeys, and L. Van Gool. Night-to-Day Image Translation for Retrieval-based Localization. *International Conference on Robotics and Automation (ICRA), arXiv preprint arXiv:1809.09767*, 2019.

R. Arandjelović and A. Zisserman. Three things everyone should know to improve object retrieval. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2911–2918, 2012.

R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5297–5307, 2016.

M. Aubry, B. C. Russell, and J. Sivic. Painting-to-3D Model Alignment via Discriminative Visual Elements. *ACM Transactions on Graphics (TOG)*, 33(2):14, 2014.

S. Bae, A. Agarwala, and F. Durand. Computational Rephotography. *ACM Transactions on Graphics*, 29(3):1–15, 2010.

V. Balntas, E. Riba, D. Ponsa, and K. Mikolajczyk. Learning Local Feature Descriptors with Triplets and Shallow Convolutional Neural Networks. *British Maschine Vision Conference (BMVC)*, 1(2):333, 2016.

M. Bansal. *"Disparate View Matching"*. PhD thesis, University of Pennsylvania, 2015.

M. Bansal and K. Daniilidis. Joint Spectral Correspondence for Disparate Image Matching. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2802–2809, 2013.

H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded Up Robust Features. *European Conference on Computer Vision (ECCV)*, pages 404–417, 2006.

A. Becker and O. Vornberger. Evaluation of Feature Detectors, Descriptors and Match Filtering Approaches for Historic Repeat Photography. *Scandinavian Conference for Image Analysis (SCIA)*, 2019.

R. Brunelli. *Template Matching Techniques in Computer Vision: Theory and Practice*. John Wiley & Sons, 2009.

A. Bursuc, G. Tolias, and H. Jégou. Kernel Local Descriptors with Implicit Rotation Matching. *International Conference on Multimedia Retrieval (ACM)*, pages 595–598, 2015.

M. Calonder. BRIEF: Binary Robust Independent Elementary Features. *European Conference on Computer Vision (ECCV)*, pages 778–792, 2010.

J. Campi. *Civil War Battlefields Then and Now*. Thunder Bay Press, 2008.

J. Canny. A Computational Approach to Edge Detection. *Transactions on pattern analysis and machine intelligence*, pages 679–698, 1986.

R. Chen and C. Gotsmann. Generalized As-Similar-As-Possible Warping with Applications in Digital Photography. *Computer Graphics Forum*, 35(2):81–92, 2016.

Y. Chen, W. Li, X. Chen, and L. V. Gool. Learning Semantic Segmentation from Synthetic Data: A Geometrically Guided Input-Output Adaptation Approach. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1841–1850, 2019.

O. Chum and J. Matas. Matching with PROSAC - Progressive Sample Consensus. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 220–226, 2005.

F. E. Clements. *Research methods in ecology*. Lincoln, Neb. The University Publishing Company, 1905.

M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. *Conference on computer vision and pattern recognition (CVPR)*, pages 3213–3223, 2016.

M. Cummins and P. Newman. FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance. *The International Journal of Robotics Research*, 27(6):647–665, 2008.

R. Diederichsen. Designing and Implementing a Rephotography Application for iOS. Bachelor thesis, University of Osnabrück, 2015.

J. Dong and S. Soatto. Domain-Size Pooling in Local Descriptors: DSP-SIFT. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5097–5106, 2015.

D. H. Douglas and T. K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2):112–122, 1973.

R. O. Duda and P. E. Hart. Use of the Hough Transformation to Detect Lines and Curves in Pictures. *Communications of the Association for Computing Machinery 15*, pages 11–15, 1972.

W. Feng, F. P. Tian, Q. Zhang, N. Zhang, L. Wan, and J. Z. Sun. Fine-Grained Change Detection of Misaligned Scenes with Varied Illuminations. *International Conference on Computer Vision (ICCV)*, 2015.

B. Fernando, T. Tommasi, and T. Tytelaars. Location Recognition Over Large Time Lags. *Computer Vision and Image Understanding*, 139:21–28, 2015.

M. A. Fischler and R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24(6):381–395, 1981.

S. Garg, N. Sünderhauf, and M. Milford. LoST? Appearance-Invariant Place Recognition for Opposite Viewpoints using Visual Semantics. *Proceedings of Robotics: Science and Systems XIV*, 2018.

C. Gat, A. B. Albu, D. German, and E. Higgs. A comparative evaluation of feature detectors on historic repeat photography. In *Advances in Visual Computing*, pages 701–714. Springer, 2011.

J. Gehrt. Asa Kinney Project: Warping Space More Than Time. `http://asakinneyproject.blogspot.de/2014/04/warping-space-more-than-time.html`, 2014. (accessed on January 30th, 2016).

A. Gore. *An inconvenient truth: The planetary emergency of global warming and what we can do about it.* Rodale, 2006.

F. C. Hall. Photo point monitoring handbook: Part A-field procedures. Technical report, Techn. Rep. PNW: USDA Forest Service, 2002.

C. Harris and M. Stephens. A Combined Corner and Edge Detector. *Alvey Vision Conference*, 15(50):10–5244, 1988.

A. E. Harrison. Reoccupying Unmarked Camera Stations for Geological Observations. *Geology*, 2(9):469–471, 1974.

R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, ISBN: 0521540518, second edition, 2004.

J. R. Hastings and R. M. Turner. *The Changing Mile: An Ecological Study of Vegetation Change with Time in the Lower Mile of an Arid and Semiarid Region.* University of Arizona Press, 1965.

G. Hattersly-Smith. The symposium on glacier mapping. *Canadian Journal of Earth Sciences*, 3:737–743, 1966.

D. C. Hauagge and N. Snavely. Image Matching Using Local Symmetry Features. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 206–213, 2012.

A. Irschara, C. Zach, J.-M. Frahm, and H. Bischof. From Structure-from-Motion Point Clouds to Fast Location Recognition. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2599–2606, 2009.

R. Jensen, A. Dahl, G. Vogiatzis, E. Tola, and H. Aanæs. Large Scale Multi-view Stereopsis Evaluation. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 406–413, 2014.

Y. Ke and R. Sukthankar. PCA-SIFT: A more distinctive representation for local image descriptors. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2, 2004.

I. Kemelmacher-Shlizerman, Shechtman E., R. Garg, and S. M. Seitz. Exploring Photobios. *ACM Transactions on Graphics (TOG)*, 30(4), 2011.

A. Kirillov, K. He, R. Girshick, C. Rother, and R. Dollár. Panoptic Segmentation. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9404–9413, 2019.

M. Klett. *Third View, Second Sights: a Rephotographic Survey of the American West.* Museum of New Mexico, 2004.

M. Klett, E. Manchester, and J. Verburg. *Second View: The Rephotographic Survey Project.* University of New Mexico Press, 1984.

N. Kobyshev, H. Riemenschneider, and L. Van Gool. Matching Features Correctly through Semantic Understanding. *Second International Conference on 3D Vision*, 1:472–479, 2014.

M. Köhler. Entwicklung einer Smartphone-App zur Produktion von deckungsgleichen Vorher-Nachher-Bildern. Bachelor thesis, University of Osnabrück, 2014.

P. Krähenbühl and V. Koltun. Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials. *Advances in Neural Information Processing Systems*, pages 109–117, 2011.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in neural information processing systems*, pages 1097–1105, 2012.

K. T. Lee, S. J. Lou, and B. Y. Chen. Rephotography Using Image Collections. *Computer Graphics Forum*, 30(7):1895–1901, 2011.

S. Leutenegger, M. Chli, and R. Y. Siegwart. BRISK: Binary Robust Invariant Scalable Keypoints. *International Conference on Computer Vision (ICCV)*, pages 248–2555, 2011.

K. Levenberg. A Method for the Solution of Certain Non-Linear Problems in Least Squares. *Quarterly of Applied Mathematics*, 2(2):164–168, 1944.

D. Levere, B. Yochelson, and P. Goldberger. *New York Changing: Revisiting Berenice Abbott's New York*. Princeton Architectural Press, 2004.

G. Levi and T. Hassner. LATCH: Learned Arrangements of Three Patch Codes. *Conference on Applications of Computer Vision (WACV)*, pages 1–9, 2016.

Y. Li, N. Snavely, D. Huttenlocher, and P. Fua. Worldwide Pose Estimation Using 3D Point Clouds. *European Conference on Computer Vision (ECCV)*, pages 15–29, 2012.

G. Lin, A. Milan, C. Shen, and I. Reid. RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

C. Liu, L. C. Chen, F. Schroff, H. Adam, W. Hua, A. L. Yuille, and L. Fei-Fei. Auto-DeepLab: Hierarchical Neural Architecture Search for Semantic Image Segmentation. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 82–92, 2019.

H. Liu, J. Zhang, J. Zhu, and S. C. H. Hoi. DeepFacade: A deep learning approach to facade partsing. *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2301–2307, 2017.

Z. Liu and R. Marlet. Virtual Line Descriptor and Semi-Local Matching Method for Reliable Feature Correspondence. *British Maschine Vision Conference*, pages 16–1, 2012.

D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

S. M. Lowry, M. J. Milford, and G. F. Wyeth. Transforming Morning to Afternoon Using Linear Regression Techniques. *International Conference on Robotics and Automation (ICRA)*, pages 3950–3955, 2014.

S. M. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford. Visual Place Recognition: A Survey. *Transactions on Robotics*, 32(1):1–19, 2016.

F. Maiwald, K. Barthel, J. Bruschke, K. Friedrichs, C. Kröber, S Münster, and F Niebling. Research and Communication of Urban History in 4D Using Historical Photographs - A Status Report of the Research Group UrbanHistory4D. *Euro-Mediterranean Conference*, pages 261–270, 2018a.

F. Maiwald, D. Schneider, F. Henze, S. Münster, and F. Niebling. Feature Matching of Historical Images based on Geometry of Quadrilaterals. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 42(2), 2018b.

F. Maiwald, J. Bruschke, C. Lehmann, and F. Niebling. A 4D information system for the exploration of multitemporal images and maps using photogrammetry, Web technologies and VR/AR. *Virtual Archaeology Review*, 10, 2019.

H. E. Malde. Geologic Bench Marks by Teresstrial Photography. *US Geological Survey*, 1(2): 193–206, 1973.

R. Martin-Brualla, D. Gallup, and S. M. Seitz. Time-lapse Mining from Internet Photos. *ACM Transactions on Graphics (TOG)*, 34(4):62, 2015.

J. Mata, O. Chum, M. Urban, and T. Pajdla. Robust Wide-Baseline Stereo from Maximally Stable Extremal Regions. *Image and Vision Computing*, 22(10):761–767, 2004.

K. Matzen and N. Snavely. Scene Chronology. *European Computer Vision Conference (ECCV)*, pages 615–630, 2014.

I. Melekhov, J. Ylioinas, J. Kannala, and E. Rahtu. Relative Camera Pose Estimation Using Convolutional Neural Networks. *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 675–687, 2017.

J. L. Meyer and Y. Youngs. Historical Landscape Change in Yellowstone National Park: Demonstrating the Value of Intensive Field Observation and Repeat Photography. *Geographical Review*, 108.3:387–409, 2018.

K. Mikolajczyk and C. Schmid. Scale & Affine Invariant Interest Point Detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.

K. Mikolajczyk and C. Schmid. A Performance Evaluation of Local Descriptors. *Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A Comparison of Affine Region Detectors. *International Journal of Computer Vision*, 65(1-2):43–72, 2005.

M. J. Milford and G. F. Wyeth. SeqSLAM: Visual Route-Based Navigation for Sunny Summer Days and Stormy Winter Nights. *International Conference on Robotics and Automation (ICRA)*, pages 1643–1649, 2012.

L. Moisan and B. Stival. A Probabilistic Criterion to Detect Rigid Point Matches between two Images and Estimate the Fundamental Matrix. *International Journal of Computer Vision*, 57 (3):201–218, 2004.

L. Moisan, P. Moulon, and P. Monasse. Automatic Homographic Registration of a Pair of Images, with A Contrario Elimination of Outliers. *Image Processing On Line*, 2(3):56–73, 2012.

J. M. Morel and G. Yu. ASIFT: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences*, 2(2):438–469, 2009.

R. K. Moseley. Historical Landscape Change in Northwestern Yunnan, China: Using Repeat Photography to Assess the Perceptions and Realities of Biodiversity Loss. *Mountain Research and Development*, 26(3):214–219, 2006.

A. Mousavian, J. Košecká, and J.-M. Lien. Semantically Guided Location Recognition for Outdoors Scenes. *International Conference on Robotics and Automation (ICRA)*, pages 4882–4889, 2015.

M. Muja and D. G. Lowe. Fast Approximate Nearest Neighbours with Automatic Algorithm Configuration. *International Conference on Computer Vision Theory and Applications (VISAAP)*, 2009.

P. Neubert, N. Sünderhauf, and P. Protzel. Superpixel-based Appearance Change Prediction for Long-Term Navigation Across Seasons. *Robotics and Autonomous Systems*, 69:15–27, 2015.

F. Niebling, F. Maiwald, K. Barthel, and M. E. Latoschik. 4D Augmented City Models, Photogrammetric Creation and Dissemination. In *Digital Research and Education in Architectural heritage*, pages 196–212. Springer, 2018.

H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han. Large-Scale Image Retrieval with Attentive Deep Local Features. *International Conference on Computer Vision (ICCV)*, pages 3456–3465, 2017.

M. Reiss and E. Joseph. *New York Then and Now*. Thunder Bay Press, 3rd edition, 2013.

J. M. Rhemtulla, R. J. Hall, E. S. Higgs, and S. E. Macdonald. Eight years of change: vegetation in the montane ecoregion of Jasper National Park, Alberta, Canada. *Canadian Journal of Forest Research*, 32:2010–2021, 2002.

B. Romaniuk, L. Younes, and E. Bittar. 3D Rheims Reconstruction Through Ages: Robust and Invariant Postcard Matching. *Conference on Image and Vision Computing New Zealand*, pages 458–463, 2012.

E. Rosten and T. Drummond. Machine Learning for High-Speed Corner Detection. *European Computer Vision Conference (ECCV)*, pages 430–443, 2006.

G. Roth and A. Whitehead. Using Projective Vision to find Camera Positions in an Image Sequence. In *Conference on Vision Interface 2000 (VI2000)*, pages 87–94, Montreal, Canada, 2000.

P. J. Rousseeuw. Least Median of Squares Regression. *Journal of the American Statistical Association*, 79(388):871–880, 1984.

E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An Efficient Alternative to SIFT and SURF. *International Conference on Computer Vision (ICCV)*, pages 2564–2571, 2011.

J. Salick, Y. Yongping, and A. Amend. Tibetan Land Use and Change near Khawa Karpo, Eastern Himalayas. *Economic botany*, 59(4):312–325, 2005.

P. E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk. From Coarse to Fine: Robust Hierarchical Localization at Large Scale. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12716–12725, 2019.

T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, F. Kahl, and T. Pajdla. Benchmarking 6DOF Outdoor Visual Localization in Changing Conditions. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8601–8610, 2018.

A. Schaffland, O. Vornberger, and G. Heidemann. A System for the Creation, Organization, and Visualization of Repeat Photographs. *International Conference on Multimedia (ACMMM)*, 2019.

G. Schindler. *"Unlocking the Urban Photographic Record through 4D Scene Modeling"*. PhD thesis, Georgia Institute of Technology, 2010.

G. Schindler and F. Dellaert. "Probabilistic Temporal Inference on Reconstructed 3D Scenes". *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1410–1417, 2010.

G. Schindler and F. Dellaert. 4D Cities: Analyzing, Visualizing, and Interacting with Historical Urban Photo Collections. *Journal of Multimedia*, 7(2):124–131, 2012.

G. Schindler, P. Krishnamurthy, and F. Dellaert. Line-Based Structure from Motion for Urban Environments. *Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)*, pages 846–853, 2006.

J. L. Schönberger and J.-M. Frahm. Structure-from-Motion Revisited. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 2016.

J. L. Schönberger, H. Hardmeier, T. Sattler, and M. Pollefeys. Comarative Evaluation of Hand-Crafted and Learned Local Features. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1482–1491, 2017.

J. L. Schönberger, M. Pollefeys, A. Geiger, and T. Sattler. Semantic Visual Localization. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6896–6906, 2018.

E. Shechtman and M. Irani. Matching Local Self-Similarities across Images and Videos. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.

J. Shi and C. Tomasi. Good Features to Track. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600, 1994.

Y. B. Shi, F.P. Tian, D. Miao, and W. Feng. Fast and Reliable Computational Rephotography on Mobile Device. *International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2018.

A. Shrivastava, T. Malisiewicz, A. Gupta, and A. A. Efros. Data-driven Visual Similarity for Cross-domain Image Matching. *ACM Transactions on Graphics (TOG)*, 30(6):154, 2011.

H.-Y. Shum and S. B. Kang. Review of Image-Based Rendering Techniques. *International Conference on Visual Communications and Image Processing (VCIP)*, pages 2–13, 2000.

E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer. Discriminative Learning of Deep Convolutional Feature Point Descriptors. *International Conference on Computer Vision (ICCV)*, pages 118–126, 2015.

K. Simonyan and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *International Conference on Learning Representations (ICLR)*, 2015.

J. M. Skovlin and J. W. Thomas. Interpreting Long-Term Trends in Blue Mountain Ecosystems from Repeat Photography. Technical report, Gen. Tech. Rep. PNW-GTR-315. Portland, OR: U.S. Department of Agriculture, Forest Service, Pacific Northwest Research Station. 114 p, 1995.

T. Smith. Repeat Photorgraphy as a Method in Visual Anthropology. *Visual Anthropology*, 20 (2-3):179–200, 2007.

N. Snavely, S. M. Seitz, and R. Szeliski. Photo Tourism: Exploring Photo Collections in 3D. *ACM transactions on graphics (TOG)*, 25(3):835–846, 2006.

M. Steinkamp. Entwurf und Implementation einer Android-App zur Unterstützung der Rephotographie. Bachelor thesis, University of Osnabrück, 2016.

C. Strecha, W. von Hansen, L. Van Gool, P. Fua, and U. Thoennessen. On Benchmarking Camera Calibration and Multi-View Stereo for High Resolution Imagery. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008.

A. Stylianou, A. Abrams, and R. Pless. Characterizing Feature Matching Performance Over Long Time Periods. *Winter Conference on Applications of Computer Vision (WACV)*, pages 892–898, 2015.

N. Sünderhauf, F. Dayoub, S. Shirazi, B. Upcroft, and M. Milford. On the Performance of ConvNet Features for Place Recognition. *International Conference on Intelligent Robots and Systems (IROS)*, 2015a.

N. Sünderhauf, S. Shirazi, A. Jacobson, F. Dayoub, E. Pepperell, B. Upcroft, and M. Milford. Place Recognition with ConvNet Landmarks: Viewpoint-Robust, Condition-Robust, Training-Free. *Proceedings of Robotics: Science and Systems XII*, 2015b.

H. Taira, M. Okutomi, T. Sattler, M. Cimpoi, M. Pollefeys, J. Sivic, T. Pajdla, and A. Torii. InLoc: Indoor Visual Localization with Dense Matching and View Synthesis. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7199–7209, 2018.

thirdview.org. Third View: A Rephotographic Survey of the American West. `http://www.thirdview.org`. (accessed on January 7th, 2020).

C. Toft, C. Olsson, and F. Kahl. Long-term 3d Localization and Pose from Semantic Labellings. *International Conference on Computer Vision (ICCV)*, pages 650–659, 2017.

C. Toft, E. Stenborg, L. Hammarstrand, L. Brynte, M. Pollefeys, T. Sattler, and F. Kahl. Semantic Match Consistency for Long-Term Visual Localization. *European Conference on Computer Vision (ECCV)*, pages 383–399, 2018.

E. Tola, V. Lepetit, and P. Fua. An Efficient Dense Descriptor Applied to Wide Baseline Stereo. *Transactions on Pattern Analysis and Machine Intelligence*, 32(5):815–830, 2010.

A. Torii, R. Arandjelović, J. Sivic, M. Okutomi, and T. Pajdla. 24/7 Place Recognition by View Synthesis. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1808–1817, 2015.

H. P. Trivedi and S. A. Lloyd. The Role of Disparity Gradient in Stereo Vision. *Perception*, 14 (6):685–690, 1985.

usgs.gov. Northern Rocky Mountain Science Center Repeat Photography Project. `https://www.usgs.gov/centers/norock/science/repeat-photography-project?qt-science_center_objects=0#qt-science_center_objects`. (last accessed on January 7th, 2020).

C. Valgren and A. J. Lilienthal. SIFT, SURF & Seasons: Appearance-based Long-term Localization in Outdoor Environments. *Robotics and Autonomous Systems*, 58(2):149–156, 2010.

J. M. von Behren. Entwurf und Implementation einer iOS-App für das re.photos-Portal. Bachelor thesis, University of Osnabrück, 2017.

J. Walker and J. Leib. Revisiting the Topia Road: Walking in the Footsteps of West and Parsons. *Geographical Review*, 92(4):555–581, 2002.

Z. Wang, B. Fan, and F. Wu. Local Intensity Order Pattern for Feature Description. *International Conference on Computer Vision (ICCV)*, pages 603–610, 2011.

R. H. Webb. *Grand Canyon, a Century of Change: Rephotography of the 1889-1890 Stanton Expedition*. University of Arizona Press, 1996.

S. Weber. Aufbau einer interaktiven Internetplattform als Django-Projekt zur Erstellung und Veröffentlichung von Vorher-Nachher-Bildpaaren. Bachelor thesis, University of Osnabrück, 2015.

A. R. Widya, A. Torii, and M. Okutomi. Structure from Motion using Dense CNN Features with Keypoint Relocalization. *IPSJ Transactions on Computer Vision and Applications*, 10 (1):6, 2018.

R. Wolfe. Modern to historical image feature matching. `http://robbiewolfe.ca/programming/honoursproject/report.pdf`, 2013. (accessed on January 7th, 2020).

K. M. Yi, E. Trulls, V. Lepetit, and P. Fua. LIFT: Learned Invariant Feature Transform. *European Conference on Computer Vision (ECCV)*, pages 467–483, 2016.

Z. Zhang. A Flexible New Technique for Camera Calibration. *Transactions on Pattern Analysis and Machine Intelligence*, 22:1330–1334, 2000.

H. Zhou, T. Sattler, and D. W. Jacobs. Evaluating Local Features for Day-Night Matching. *European Conference on Computer Vision (ECCV)*, pages 724–736, 2016.

C. Zitnick. Binary Coherent Edge Descriptors. *European Conference on Computer Vision (ECCV)*, pages 170–182, 2010.

Acknowledgements

First of all, I thank my supervisor Prof. Dr. Oliver Vornberger for his support and advice during the composition of this thesis. Second I would like to thank Prof. Dr. Heipke who agreed to provide the second opinion on this thesis. Furthermore, I thank Axel Schaffland, Rasmus Diederichsen and Sören Weber for their helpful discussions and their contributions in the context of the re.photos portal. For general support, advice and proofreading my drafts I would like to thank my friends and colleagues Mathias Menninghaus, Nils Haldenwang, Laura Hembrock, Elisaweta Ossovski and Sven Klecker as well as Friedhelm Hofmeyer who provided professional support on all technical issues. Finally, I thank my husband Stefan Becker and my parents Heidemarie and Bernhardt Häuser for their encouragement and continuous support, especially during the stressful phases of this work.