



Article

Identifying Historic Buildings over Time through Image Matching

Kyriaki A. Tychola ^{1,*}, Stamatis Chatzistamatis ^{1,2}, Eleni Vrochidou ¹, George E. Tsekouras ²
and George A. Papakostas ^{1,*}

¹ MLV Research Group, Department of Computer Science, International Hellenic University, 65404 Kavala, Greece

² Department of Cultural Technology and Communications, University of the Aegean, 81100 Mytilene, Greece

* Correspondence: kytzcho@cs.ihu.gr (K.A.T.); gpapak@cs.ihu.gr (G.A.P.); Tel.: +30-2510462361 (G.A.P.)

Abstract: The buildings in a city are of great importance. Certain historic buildings are landmarks and indicate the city's architecture and culture. The buildings over time undergo changes because of various factors, such as structural changes, natural disaster damages, and aesthetic interventions. The form of buildings in each period is perceived and understood by people of each generation, through photography. Nevertheless, each photograph has its own characteristics depending on the camera (analog or digital) used for capturing it. Any photo, even depicting the same object, is impossible to capture in the same way in terms of illumination, viewing angle, and scale. Hence, to study two or more photographs depicting the same object, first they should be identified and then properly matched. Nowadays, computer vision contributes to this process by providing useful tools. In particular, for this purpose, several feature detection and description algorithms of homologous points have been developed. In this study, the identification of historic buildings over time through feature correspondence techniques and methods is investigated. Especially, photographs from landmarks of Drama city, in Greece, on different dates and conditions (weather, light, rotation, scale, etc.), were gathered and experiments on 2D pairs of images, implementing traditional feature detectors and descriptors algorithms, such as SIFT, ORB, and BRISK, were carried out. This study aims to evaluate the feature matching procedure focusing on both the algorithms' performance (accuracy, efficiency, and robustness) and the identification of the buildings. SIFT and BRISK are the most accurate algorithms while ORB and BRISK are the most efficient.

Keywords: computer vision; feature extraction; building identification; culture heritage; architectural heritage; image descriptor; image matching; image analysis



Citation: Tychola, K.A.; Chatzistamatis, S.; Vrochidou, E.; Tsekouras, G.E.; Papakostas, G.A. Identifying Historic Buildings over Time through Image Matching. *Technologies* **2023**, *11*, 32. <https://doi.org/10.3390/technologies11010032>

Academic Editor: Gwanggil Jeon

Received: 14 January 2023

Revised: 4 February 2023

Accepted: 11 February 2023

Published: 17 February 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

A city is continuously evolving, and this implies large and unforeseen changes. As part of the urban landscape, historic buildings are the epitome and reflection of civilization. Over time, the urban net grows rapidly, and this entails large and unexpected spatial-temporal changes in the buildings. Some buildings have historical importance and are considered landmarks of a city [1]. Architectural heritage is a wealth of cultural heritage expression and an invaluable testimony to the past. Hence, it should be protected [2]. This is achieved with new technologies. Nowadays, the contribution of computer vision with various tools available plays a catalytic role in building protection. For instance, a comparative study of the building evolution over different periods has positive effects on the cultural heritage of the city and for anyone interested. In addition, comparing images of historic buildings helps those in charge of better decision-making in the management of heritage [3]. For this purpose, various suitable matching techniques are applied aiming at the correct correspondence between images and their further processing [4].

An image is a result of a camera recording and is a way to capture reality. The images, which capture an entity over time, such as a building, were not captured under the same

conditions. Therefore, for their comparison, feature extraction and matching techniques should be applied. Specifically, for feature extraction and matching from two or more images of the same scene but from different viewing angles and captured from the same or different camera, homologous point detection techniques are used. This is an important task in image analysis [5]. Especially, it is the primary preprocessing image step for further processing, and it has been widely applied in various computer vision applications, such as pattern recognition, robot navigation, image stitching and mosaicking [6], visual odometry [7], pose estimation, object classification, and 3D reconstruction in the case of 3D images [8,9]. In the case of buildings, this procedure is of major importance, as in a further analysis of the images they can be compared and identify possible lesions and deformations. To achieve strong feature matching, the invariant properties of the images should be utilized in order that the extracted features not vary with respect to lighting changes, scale, position, rotation, and viewing angle [10].

Feature extraction from images is achieved by feature detectors, which detect feature invariants when the image undergoes different transformations. The term ‘invariant feature’ refers to those features that remain invariant when rotation, scaling, illumination, and affine transformation are applied. Detection is achieved by scaling the image to extract distinct features across various scales of objects to identify. After this process, various descriptors are applied to describe the features extracted with repeatability, compatibility, accuracy, and efficient representations, which are also invariant to scale, rotation, affine transformation, occlusion, and illumination [2,11–13].

After feature detection and description follows the feature matching. The matching procedure finds and matches identical points between image pairs by calculating the displacement from the changes in the pixels [14]. There are two main image-matching methods: the area-based methods, where detectors are applied to find the similarity of the pixels between source and target images followed by optimization algorithms [15–18], and the feature-based methods, where the features are extracted directly from images without calculating the intensity values. The latter category is suitable for images with complex geometric distortions and lighting changes [17,18]. Historical images of buildings are an important source of information for researchers. Finding images of specific objects, visually comparing different constructions, or estimating proportions, is important. These tasks are largely related to metadata (relative position, orientation of images, and download time), and since the quality of metadata varies, it poses a problem in detection and identification of entities in an image [19]. To date, there are reliable image-matching techniques for matching vector- or binary-based images to modern images. Nevertheless, they present inaccuracies in historical images [20–22].

Although in the literature, there are many studies on building management and conservation, such as building disaster prediction [23], there are very few studies about feature-matching methods and techniques for historical building 2D images. For instance, Agarwal et al. [24] have presented edge-detection techniques for matching images, based on graphs. Surapog et al. [25] proposed the fusion of historical and modern photographs with a database-based indexing technique; however, their original purpose was not the matching. Heider and Whitehead [20] applied vector-based, binary-based, and hybrid techniques, such as ORB detector/SURF descriptor, on historical and modern buildings of the same landmarks to examine the correspondence problem. Hybrid techniques have proved the most efficient, whereas in the same period they used feature descriptors based on the gradient of two historical images that converted them into a matrix. The results were incorrect [26]. Wolf et al. [27] proposed an innovative approach for feature matching using image regions. The expectation was that using regions instead of corners and edges would lead to greater precision in the matching process. The initial results were mixed, and no concrete solution was found. Wu et al. [28] have proposed a matching methodology for historical buildings based on contours. They based it on the Canny edge-detection algorithm and replaced the Sobel filter with a modified Scharr filter, and they automatically adjusted the local limit, instead of using the Canny algorithm settings. Their algorithm was

sensitive to multidirectional gradient change but was very efficient at detailed building corners. Kabir S. R. et al. [2] proposed four computational methods to feature detection (Canny edge detection, Hough line transform, find contours, and Harris corner detector) on historic and modern buildings. Then, they evaluated their algorithms' performances, concluding that these detectors were best suited for this purpose. Samaun Hasan et al. [29] applied the Canny edge detector to a dataset of modern and old Indian buildings and then designed a neural CNN model able to distinguish two different time periods (Sultans and Mongols). Maiwald F. et al. [30] used exclusively geometrical features and semantic constraints (windows, material, and overlays) to match two or more images. As long as strong features were taken into account, the linear structures of objects were easily detected as quadrilaterals. Yue L. and Zheng X. [31] proposed a method for the distorted images of buildings but not of historical ones. They applied the TILT algorithm to correct the image, and an automated detection method to low-textured photographs, while for image matching the ORB algorithm was used to remove the outliers. Si, L. et al. [32] implemented a genetic image-matching algorithm for associating two building images but not historical ones to solve the problem of optimization and proposed techniques for quickly associating homologous points. Edward J. and Yang G.Z. [33] applied the RANSAC algorithm to pairs of a building's images in changing environments for learning invariant features, during the day and different seasons. The authors emphasized the geometry of similar buildings and the calculation of distances and corners between all the matched features. Finally, Avrithis, Y. and Toliás, G. [34] introduced a methodology of histogram pyramids based on Hough voting, where the votes were derived from the single feature matches, for accelerated large-scale image retrieval. It is noted that this investigation was not only limited to buildings. In addition, the authors applied the SURF algorithm to extract and describe the image features.

This study attempts to investigate the feature extraction and matching from various aspects such as comparing historical buildings over time between them, as well as with buildings of other architectures. In addition, it studies the performance of three algorithms and the possible changes on buildings over different time periods. More specifically, this work is focused on the employment of well-established image-matching algorithms to the problem of historical buildings' identification over time, given their indisputable performance and considering the inherent limited data available for the problem under study. To accomplish that task, three algorithms are compared, namely, the SIFT [35], the ORB [36], and the BRISK [37] algorithm. Alternative methods such as deep learning could be investigated; however, they are in need of large datasets, which in the case of historical buildings over time are challenging and time-intensive to collect.

The current research focuses on studying the correspondence problem through experiments, related to the identification of historical buildings in Drama city, Greece. Feature extraction and matching techniques on 2D images were applied, which were captured in different conditions of lighting, viewing angle, scale, season, weather conditions, and 24 h period. Then, the results were evaluated by focusing on both the performance and reliability of used algorithms, and any changes on buildings were reflected through the results.

The rest of the paper is organized as follows: Section 2 presents the motivation and contribution of this work. Section 3 presents and provides our research strategy, including a theoretical background of features extraction and image matching. In addition, the correspondence problem is highlighted and its importance in the computer vision is discussed. Section 4 describes the dataset and presents the simulation experiments, explaining the parameters that were implemented. Section 5 illustrates the experimental results. Finally, Section 6 report the discussion and conclusions of this study.

2. Motivation and Contribution

Our motivation for conducting a comprehensive survey on identifying historic buildings over time through image matching is twofold. First, the significance of the subject. Simply stated, the ability of algorithms to extract and match features between two images

is the essence of most computer vision tasks. Second, the previous reviews on the subject (see next section) carry out experiments with traditional algorithms, either focusing on their performance on historic and/or modern buildings or aiming to identify and retrieve the images through databases (image retrieval).

Undoubtedly, the previous works offer a significant contribution. Most of them analyze the algorithm performance for feature matching at the level of feature extracted from two or more images. Nevertheless, none of them provides a comprehensive study gathering, presenting, and analyzing in detail both the algorithm performance and identifying historic buildings over time among them and with other buildings, through experiments. This encourages us to carry out a thorough investigation through experiments studying all factors and parameters that are crucial in identifying buildings. Hence, the contribution and innovation of our work lie in that we present a comprehensive analysis of classical detectors and description algorithms such as SIFT, ORB, and BRISK, we conduct experiments on image pairs of the same building and on different buildings with the best parameters of these algorithms, and we analyze their capabilities and limitations. In addition, we provide solutions and identify open issues and research trends, thereby providing directions for future work. Hence, our unique contribution is a focused and detailed analysis of identifying historic buildings over time through image matching.

Moreover, we try to answer various research questions about identifying historic buildings over time through image matching. The research questions are summarized below:

- Does time affect the identity of the building?
- To what extent is the validity of the identification affected?

3. Literature Analysis

The research methodology that has been followed in this paper included an initial search on Scopus with the following query: TITLE-ABS-KEY (feature AND detectors AND descriptors). This search, conducted in November 2022, yielded 1832 papers, but only journal articles, conference papers, and book chapters were selected. To pursue a more exhaustive exploration of the literature we went beyond Scopus and coupled it with a search in Google Scholar, using the same terms. Then, our probing branched out in two parts. First, we conducted a series of secondary, more detailed Google Scholar searches, using terms specific to the phrases “feature matching methods” and “feature extraction algorithms”. Then, we sought and gathered any relevant works referenced in the papers from our primary Scopus search, examined and properly evaluated them. This wide-reaching reconnaissance brought 87 papers under our scrutiny, which we screened for relevancy. Figure 1 shows the percentage of each type of publication in the papers selected, while Figure 2 shows the progress of the published papers per year. Early in our exploration of the bibliography, we noticed that this field is quite old. The first publications date back to 1983, with the volume of research steadily increasing over the following years.

According to the above chart, the studies start “timidly” only in 1983, and for the first 9 years, there are few posts on this topic. From 2004 to 2016, researchers’ interest seems to be increased. From 2016 to date, a slight drop in publications is noted. This phenomenon can be explained by the fact that deep learning methods have been gaining more and more ground in recent years. The correspondence problem is still being investigated and seems to be attracting interest from the scientific community.

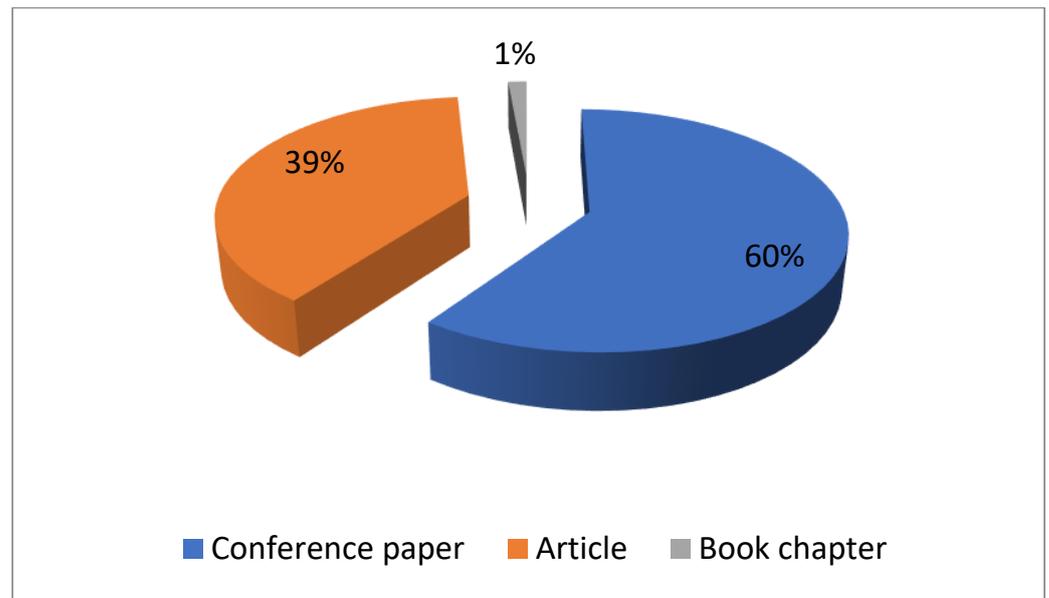


Figure 1. The pie chart illustrates the proportion of papers from different types of publications.

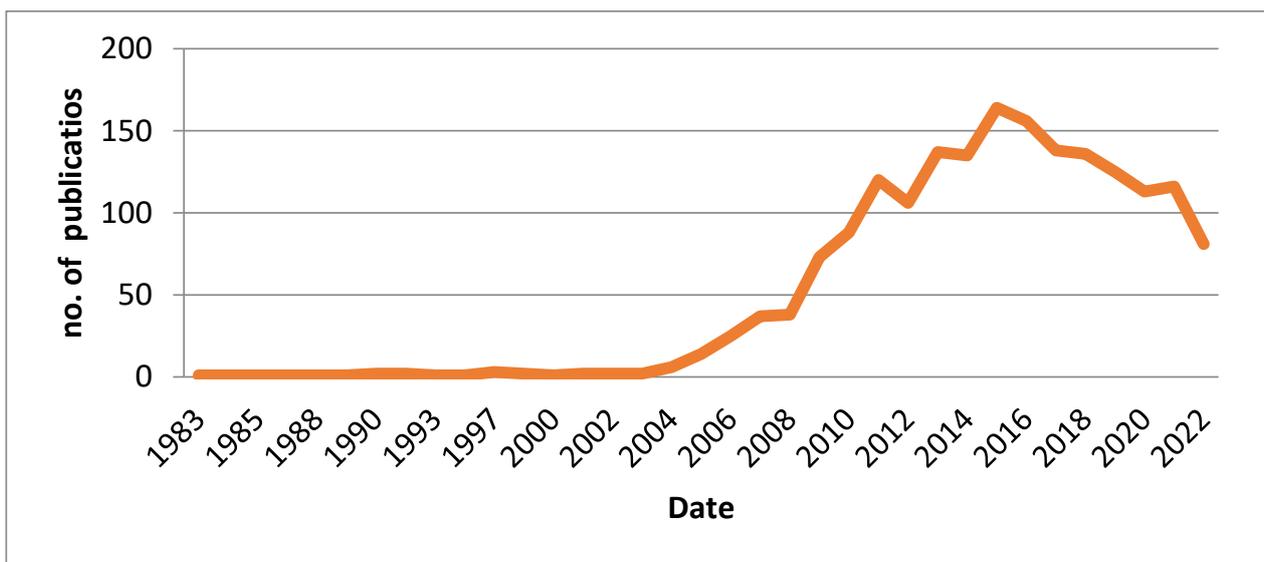


Figure 2. Number of relevant publications per year, about feature detectors and descriptors (statistics from November 2022).

4. Materials and Methods

In image processing, feature extraction is the process of transforming raw data into numeric attributes that can be further processed, preserving the information in the original dataset [38–42]. This procedure aims to reduce the number of attributes of a dataset and to create new from existing ones [43]. The characteristics to be selected should be representative so that they contain as much information as possible about the original set of features. The usefulness of this process is to increase the accuracy of the results [44,45]. Nowadays, we have at our disposal a wide variety of images depicting the same landmarks over time, which are either captured by analog cameras with film or digital cameras. It is understood that a collection of photographs that contains the same object, e.g., a building, cannot be identical (different viewing angle, different material, various weather conditions, lighting, scale, or date). Therefore, feature matching between historic and modern images is a complex process and has constraints such as the correspondence problem.

4.1. Correspondence Problem

After feature extraction follows feature matching, which is undoubtedly the main element in various applications such as optical flow stereo vision and structure from motion (SfM) [46,47]. Feature matching refers to finding matching attributes from two similar images, based on a search-distance algorithm. One of the images is considered the source (reference image) and the other the target, and the feature-matching technique is applied either for detection or for extraction and transport of a feature from reference to target image. In particular, feature matching analyzes the topology of images, detects feature patterns, and matches with features within localized patterns. The accuracy of matching features depends on the similarity, complexity, and quality of the image. In addition, feature selection plays an important role. For this purpose, various algorithms have been developed; however, none of them can be universally accepted as to high efficiency and accuracy [48]. The idea of feature matching arose in the early 1980s, when it was realized that the human brain associates (i.e., matches) entities altogether based on their characteristics and not just the tones of hue. Matching becomes less sensitive to radiometric and geometric distortions of images and takes into account space structure information, ensuring more powerful and reliable solutions. There are two stages for matching: the first stage includes feature extraction such as points, lines, edges, corners, and regions on each image, which is linked to descriptive properties (attributes) in the form of descriptive vectors (descriptors), and the second relates to feature matching after calculating a measure of similarity between their properties. In this case, the search for homologous points is restricted to the index and to descriptive features and not to the entire image, reducing managed information and thereby increasing the computational cost. When Takeo Kanade was asked, “What are the three most important problems in computer vision?” he replied, “Correspondence, correspondence, correspondence” [49]. Successful features matching allows us to create matches between pairs of points and interpret the visual world (Figure 3).



Figure 3. Feature detection and matching homologous points.

The correspondence problem remains one open issue. Given two or more images of the same scene captured from different perspectives, the correspondence problem refers to the problem of determining the parts of an image to which parts of another image correspond, i.e., the points of features in an image are the same with another image and create homologous points. In addition, the problem intensifies further because of other factors that are mentioned above [50,51]. Figure 4 shows an example of the correspondence problem.

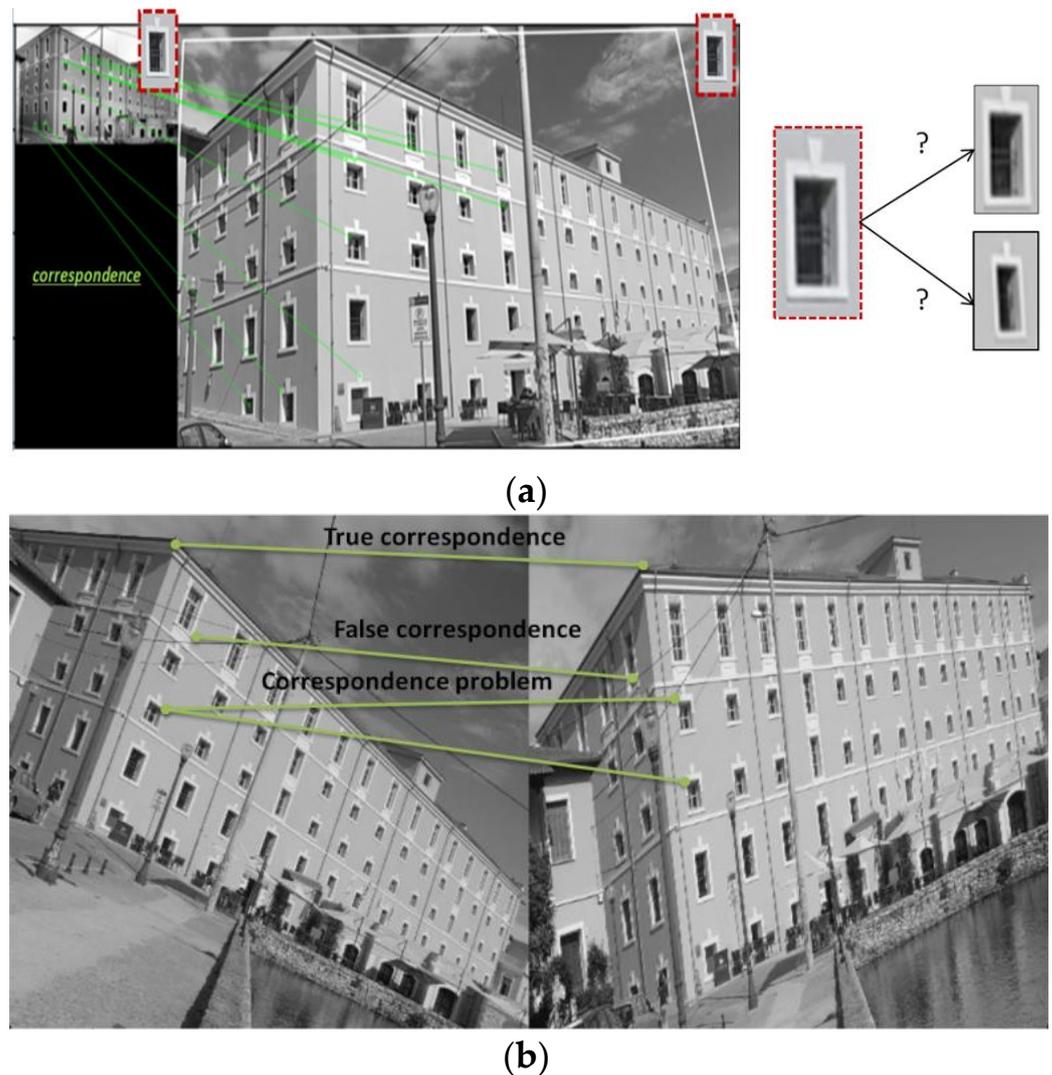


Figure 4. Correspondence problem: (a) feature matching between two images with the same building; (b) clear indication of the correspondence problem. There are false and true correspondences but in the third case (i.e., + correspondence problem), the algorithm is having trouble locating the position of the point between similar characteristics.

4.2. Feature Detectors and Descriptors for Feature Matching

Feature detectors and descriptors have been powerful tools over the last few decades in many applications in the computer vision field and pattern recognition. Feature is a piece of information extracted from an image representing a more detailed region and can be divided into *global*, which provides information about the entire image, and *local*, which focuses on a specific part of an image [52]. A feature detector to be qualified should fulfill the following criteria [5]:

- *Stability*: the locations of the features detected should be independent of different geometric transformations, scaling, rotation, translation, photometric distortions, compression errors, and noise;
- *Repeatability*: detectors should be able to detect the same features of the same scene or object repeatedly under various viewing conditions;
- *Generality*: detectors should be able to detect features that can be used in different applications;
- *Accuracy*: the feature detection should be the same accuracy localized both in image location;
- *Efficiency*: fast detection to support applications in real time;

- *Quantity*: the number of detected features should be sufficiently large, such that a reasonable number of features are detected even on small objects.

After detection follows the description of interest points. The terminology “interesting feature” includes several interchangeable terms, such as keypoint, landmark, interest point, or anchor point, all of which refer to features such as corners, edges, or patterns that can be found repeatedly with high likelihood [53]. A feature descriptor can be computed at each key point to provide more information about the pixel region surrounding the keypoint and it is a representation of transforming the located features into a new space called the feature description space, where the feature matching is more easily distinguished [54–56]. Feature description is foundational to feature matching, leading to image understanding scene analysis. The central problems in feature matching include how to determine if a feature is differentiated from other similar features, and if the feature is part of a larger object. The method of determining a feature match is critical, for many reasons such as computational cost, memory size, repeatability, accuracy, and robustness, while a perfect match is ideal. In practice, a relative match is determined by a distance function, where the incoming set of feature descriptors is compared with known feature descriptors [57,58]. Figure 5 illustrates the steps of feature matching.

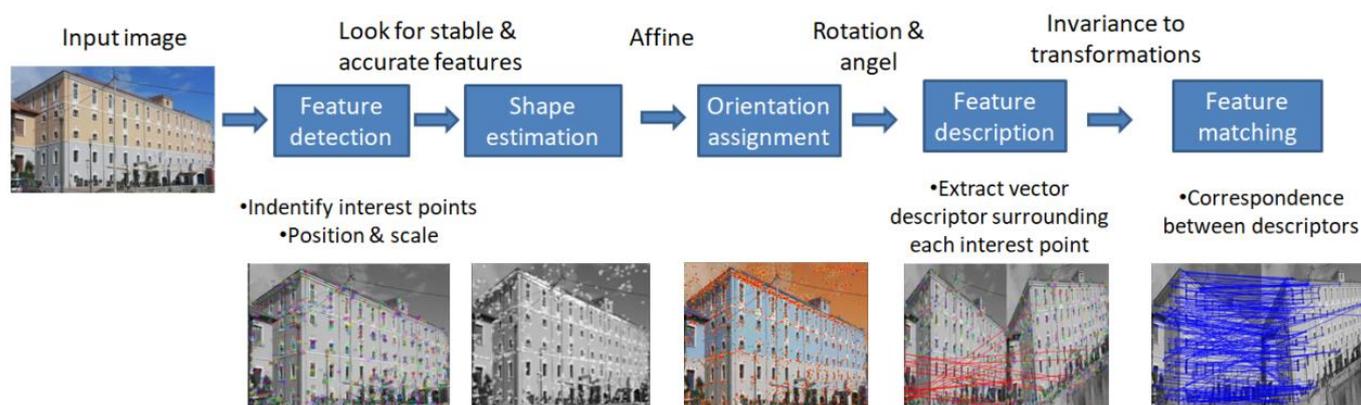


Figure 5. Feature detection and matching process.

According to Figure 5, scaling differences exist between the same images, especially in regions where the distances are close. First, the features detected, such as corners, detect interest points in a space scale (identify position and scale) around each image while subsequently local descriptions are extracted from the adjacent areas of these points. Nevertheless, feature extraction may not extract an adequate number of points of interest because of the bad texture of repeated patterns, different viewing angles, lighting, and blur [59,60]. Second, affine transformation is applied to reduce asymmetry and different scales on the axes of the two images, and their orientation is assessed, considering the different rotations of the two images around the detection point. From the above procedure is determined the characteristics’ position, scale, affine shape, and rotation. Therefore, a high-dimensional vector represents features detected, descriptors are derived independently for each image, and the matching of the two images is achieved by searching the neighborhood [55,60]. Image-matching methods can be divided into two major categories: area-based methods emphasizing the step of matching and work directly with the image intensity value, and feature-based methods that are based on feature extraction of important structures such as regions (forest and lakes), lines (area boundaries, coastlines, roads, and rivers) or points (area corners, line intersections, and high curves) [61–65]. In recent years, hybrid approaches have been applied at the same time, i.e., methods based on areas and features [66].

To identify historic buildings over time through image matching, we carry out experiments applying various classical feature detection and description algorithms such as SIFT [35], ORB [36], and BRISK [37] using the trial-and-error method for feature matching.

This section can be divided by subheadings. First, we describe our dataset and discuss the buildings' history. Second, we discuss the background of the algorithms, focusing on their parameters, and we analyze, in detail, the parameters selected for our experiments. Finally, we provide a concise comparison among algorithms.

4.3. Study Area

Drama city is part of the East Macedonia and Thrace region (Figure 6) and is located in northeastern Greece. Drama has experienced a great economic boom mainly during the Ottoman era with tobacco production and trade. This area was chosen, as it is a tourist attraction because of the various historical events over time and is considered a cultural center of great interest with milestones of various architectural orders.

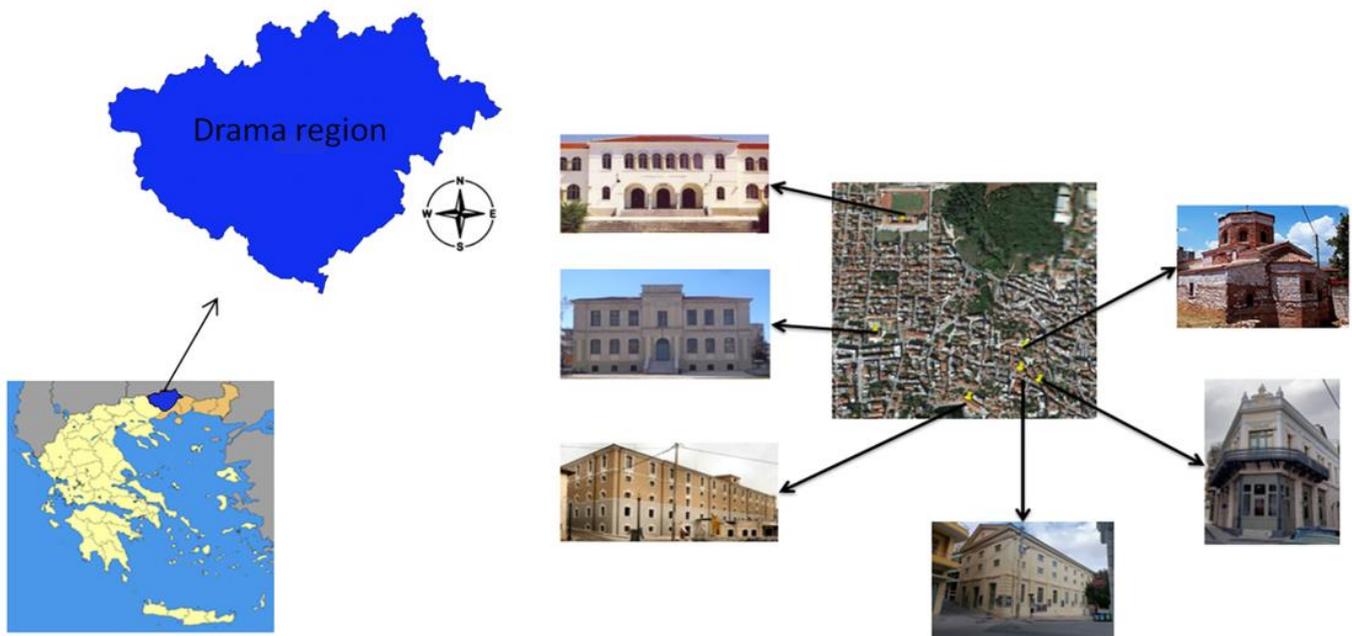


Figure 6. The study area is a part of Drama city. In the smaller map of Greece is the location of Drama, while the yellow pins represent the four landmarks that were used in this study.

4.4. Data Acquisition and Description

For the needs of the experiments, various images, with landmarks at various years, under different condition shots, and with different lenses for analog cameras and digital cameras, of Drama city in Greece were collected from the site <https://gr.pinterest.com/> (accessed on 13 January 2023) and books. In addition, we used images from personal files received on mobile phones such as Samsung A21 and Huawei P20 Lite. Furthermore, historical photo collectors have offered us some images from their collections. Note that the collection of images of historical buildings over time requires very extensive and systematic research in a large volume of files and different sources that may not be easily accessible. Finally, our dataset includes four different landmarks: tobacco industry (1990–2022), elementary school (1908–2020), Hagia Sofia church (1994–2022), and café “Eleutheria” (1940–2022), where each landmark contains 28, 7, 39, and 17 images, respectively. Figure 7 shows a sample of our dataset.

The tobacco industry was built in 1874 and belongs to I. Anastasiades and was a sample of economic development of Drama city. The school was built in 1908 on the initiative of the Metropolitan of Drama in the plans of the architect Chrysostomos Hatzimichalis. The family of Pavlos Melas offered a part of the cost, while artisans of the region aided in the construction. Nowadays, it operates as an elementary school (12th elementary school). The Church of Hagia Sophia is the oldest preserved building in Drama. It was built in the highest part of the city during the 10th century, along with the city's old

walls. The conditions for the erection of the monument are unknown, but its construction coincides with a Byzantine period of great prosperity. In addition, the church was probably dedicated to Theotokos and it was built on the ruins of a larger three-aisled church. An early Christian basilica, where debris was revealed during the landscaping work, and in terms of architecture, the temple keeps similarities with iconic monuments of Byzantine church architecture, such as the Hagia Sophia of Thessalonica and the Assumption of the Virgin Mary in Nicaea, Bithynia, Asia Minor. The historical café “Eleftheria” was built in the early 20th century (1906–1907) by the Greek community of Drama at the intersection of the present Venizelou and Kountouriotou streets. It got its name “Eleutheria” (“Freedom” in Greek) after the liberation of the city. The last few years it operated as a coffee shop, whereas, sometimes, it offers hospitality to painting exhibitions.

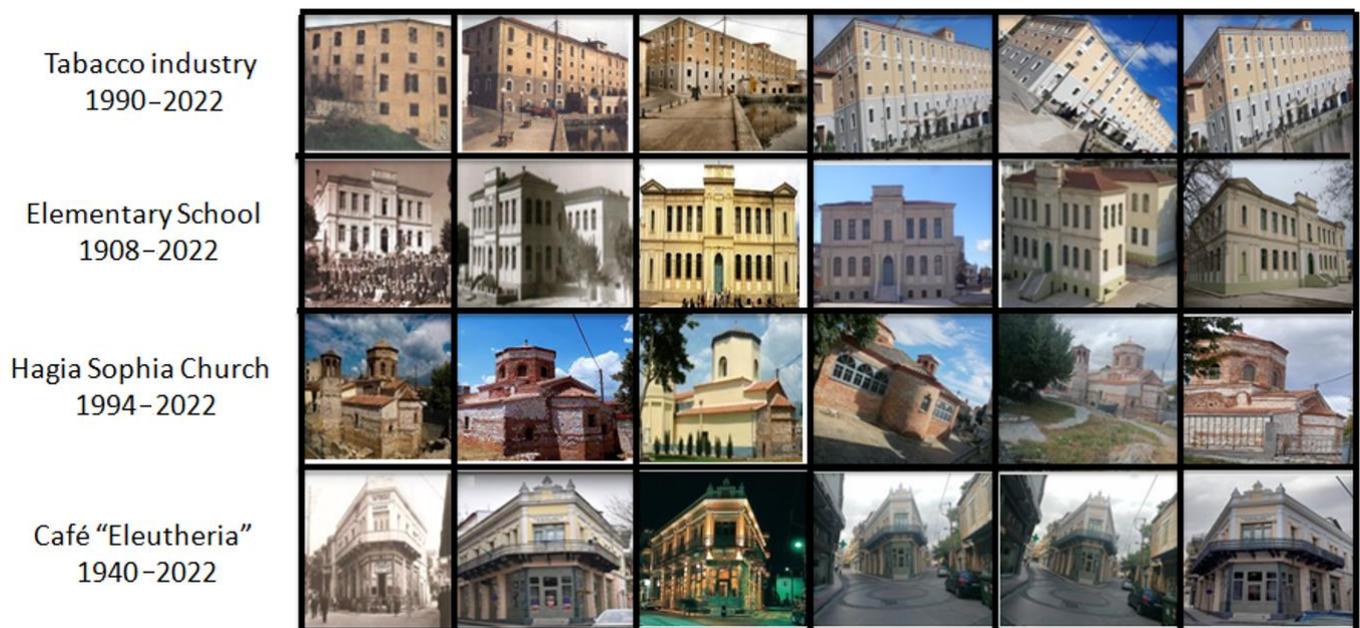


Figure 7. Landmarks samples from the dataset of Drama city.

4.5. Experiments

In this work, we carry out experiments using handcrafted matching methods to investigate feature extraction and matching for identifying historic buildings over time that the tobacco industry is the basic building of this study. Specifically, we implement a trial-and-error method to find the best parameters of algorithms that make them efficient. It should be noted here that this is not a generic approach, and thus, the extracted parameters may need to be recalculated for large deviations in the buildings’ architecture. The focus of this work is on the employment of well-established algorithms in the field of image matching to be systematically applied to the problem of historical buildings’ identification over time, given their indisputable performance and considering the limited data at our disposal. To this end, we apply classical algorithms to pairs of images, such as SIFT, ORB, and BRISK. These algorithms were selected because they present some advantages. For instance, SIFT (scale-invariant feature transform) is perfectly suitable because it detects features that are invariant to image scale and rotation. Moreover, features are robust to lighting changes, noise, and minor changes in viewpoint. ORB features are also invariant to scale and rotation, while the BRISK algorithm is robust to noise and invariant to scale and rotation. All the experiments were designed and implemented in Open CV. Figure 8 shows the workflow chart of our methodology.

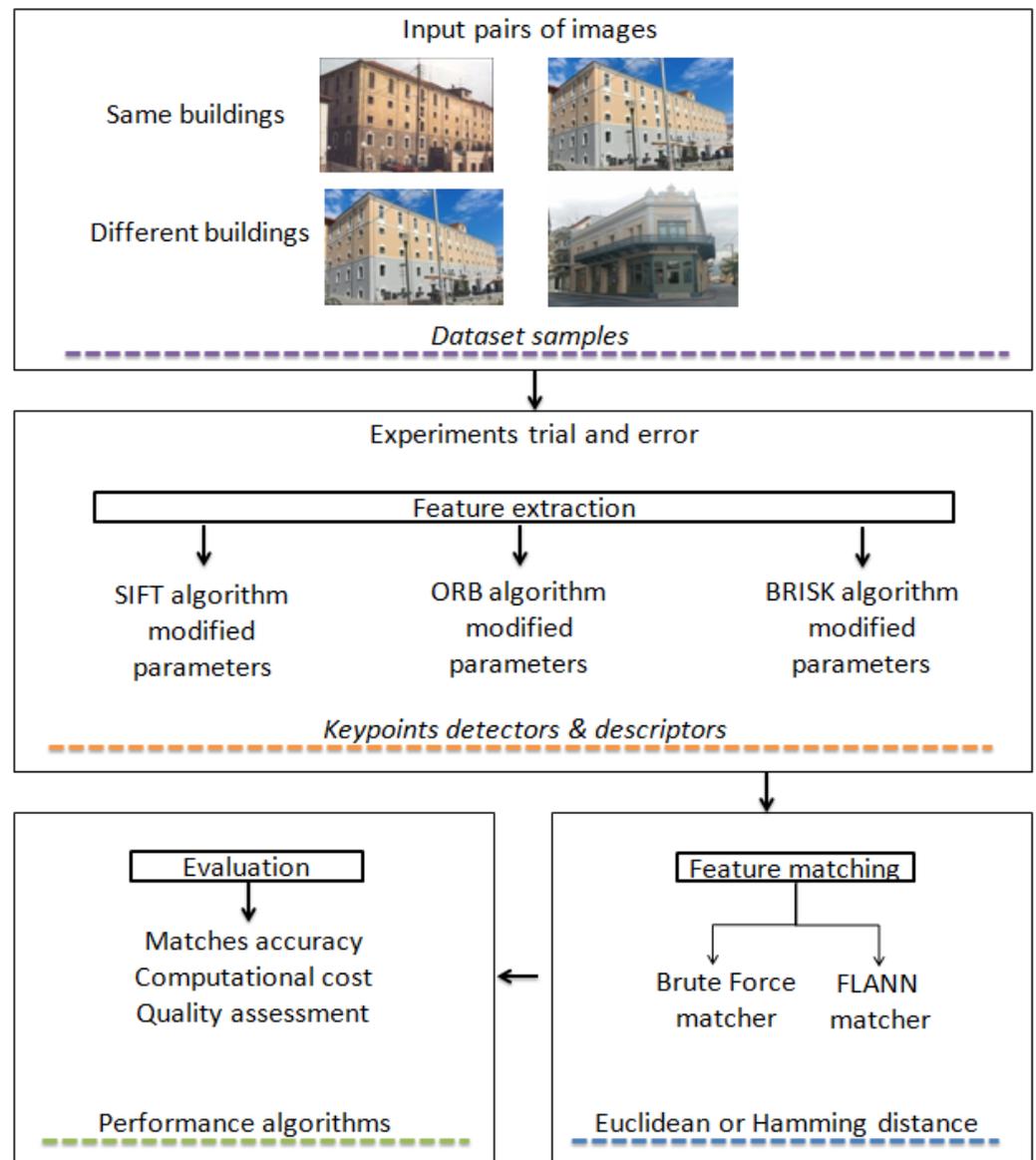


Figure 8. The flowchart of experiments.

We divided the whole procedure into four stages. Specifically, first, we input pairs of images of the tobacco industry from 1990 until 2022 (pair 9–pair 28). It should be noted that images of 2022 captured the building from different aspects, scales, and rotations. In addition, we consider that buildings from 2021 (pair 6–pair 28) and beyond are modern buildings, whereas the rest are historic (pair 1–pair 5). Besides, we compared each date of the building to all the rest, while then we tested the tobacco industry landmark with three other buildings based on the architectural orders, structure materials, and date. In the second step, we applied feature detectors and descriptions, such as SIFT, ORB, and BRISK algorithms to extract features from pairs of images. In this step, after tests, we modified the algorithm parameters in such a way as to produce the best possible results without interfering with any extra optimization method. In the third stage, we adopted the strategy of nearest neighbor distance ratio (NNDR), which was proposed by K. Mikolajczyk in [67] and D.G. Lowe in [68]. In this case, a ratio of the nearest neighbor to the second nearest neighbor is calculated for each feature descriptor, and a certain threshold ratio is set to filter out the preferred matches. We implemented the brute force matcher [68,69], i.e., a matching descriptor that compares two sets of descriptors' keypoints and produces a list of matches, and then the FLANN (FAST library for approximate nearest neighbors) matcher [70,71],

which is used in two different forms, depending on the algorithm. Finally, in the next step, the matching pairs are evaluated by various quantitative and qualitative techniques. It is understood that because of the different applications of algorithms, it is difficult to judge their performance with a unified and commonly accepted system. Hence, various indicators should be found for evaluation. Commonly, precision and best score matching for each pair of images based on the distance (Euclidean or Hamming) are evaluated as algorithm performance benchmarks [72]. The closer the homologous points between two images, the better and more valid the match is. However, since only these evaluations were not sufficient, we applied additional measures, including the computational cost.

In what follows, our evaluation measures are listed:

- The total number of keypoints from each pair of images.
- The total number and the rate of the best matches (good matches). The percentage of best matches was calculated by dividing the best matches by the smallest number of keypoints extracted from the first or second image (good matches/min no, keypoints*100).
- Precision is a performance measure of the best matches (best matches/total matches), while then, we evaluated optically the best matches, aiming to find the false positive matches.
- Effectiveness (%) measure (total matches/total keypoints) to evaluate the actual number of keypoints used for matching.

4.6. SIFT, ORB, BRISK Trial and Error Methods

4.6.1. SIFT Algorithm

SIFT (scale-invariant feature transform) algorithm has been proposed by D. G. Lowe [35] and extracts invariant features. Today, it is globally accepted and is considered the best and most accurate algorithm. The algorithm detects local features (keypoints) and localizes them on the image. Its greatest advantage is that SIFT is robustly invariant to image rotations, scale, and limited affine variations, while its main drawback is the high computational cost. Figure 9 illustrates a SIFT workflow chart.

SIFT parameters are number of features, nOctaveLayers, contrast, edge Threshold, and Gaussian sigma during the parameter test. Since the desired effect was to extract the keypoints number as large as possible, the number of features was not set with value. As the number of octavelayers increases, the number of keypoints increases; however, it only increased by one point because a further increase did not favor the increase in matches partially, i.e., during the test, with a decrease of 1, the difference is also matches and keypoints decreased. ContrastThreshold is a parameter that specifies under which threshold the values of keypoints should be discarded. To be accepted, there must be points with contrast above the threshold. Increasing the value of filtered, more and more points are discarded. In short, as the value of the parameter increases, the process tightens, but at the same time, the success rate of the mappings increases. The ideal is many keypoints and a high success rate. The edgeThreshold parameter is a limit for the ratio of two identical vectors given by filtering keypoints on edges to discard them. The higher the limit, the more difficult it is to discard possible keypoints. As the threshold value increases, the number of keypoints increases. In the context of the test with the price fluctuation, the difference is not obvious and does not offer any improvement. The success rate decreases as the value increases. Because “pass” keypoints are not completely valid, the reduction is therefore negligible. The value of Gaussian sigma depends more on the quality of the images. Since most pictures have a good resolution, it was not considered necessary to dramatically change the value of the original. Increasing the value observed a significant decrease in keypoints and a large percentage of matching that was false, while a slight value increase decreases dramatically both the matching and the number of keypoints but not proportionally. Note that if the images were clear and largely blurred, a small value would be required. Therefore, the value selected was 1 point less than the default value. Finally, we also increased the ratio.

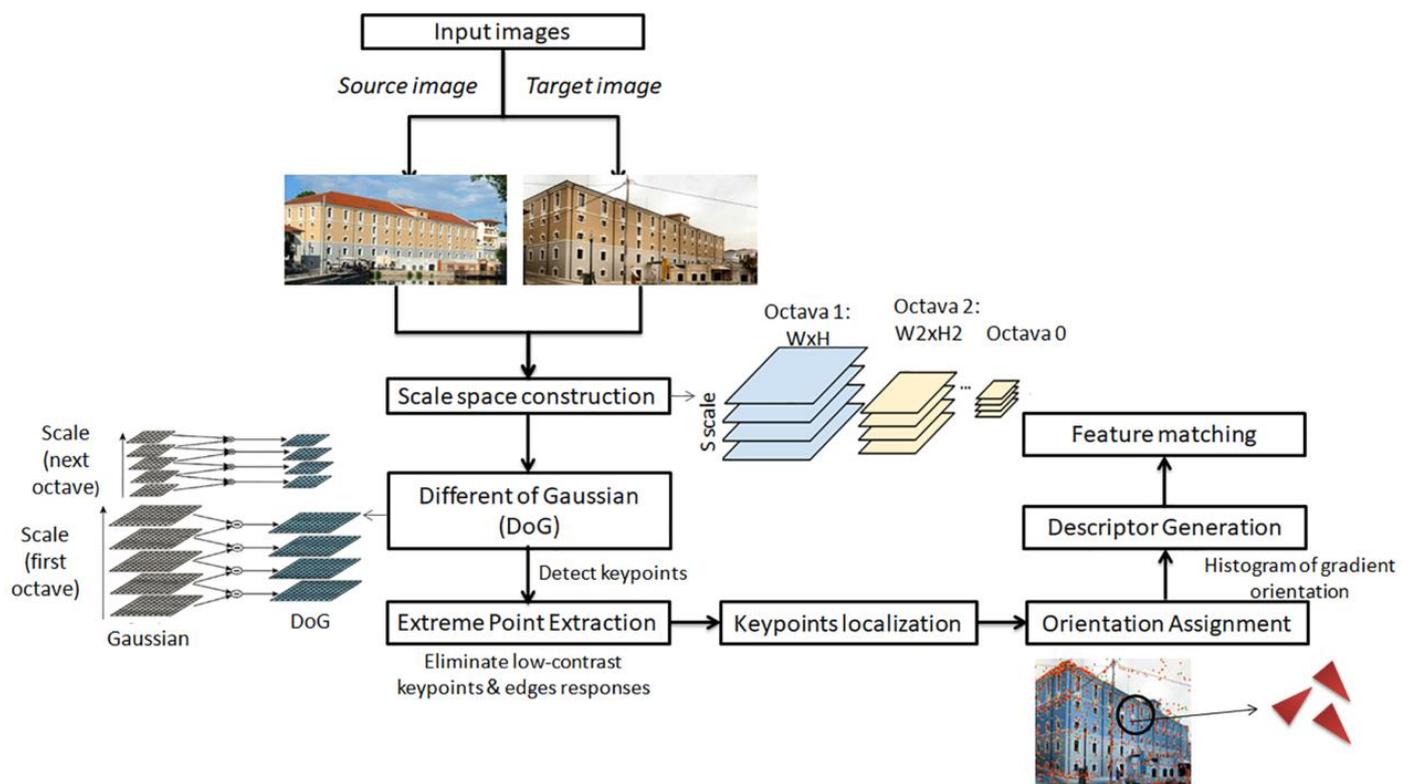


Figure 9. Workflow chart of SIFT algorithm. SIFT detector is based on the difference-of-Gaussians (DoG) operator, which is an approximation of Laplacian-of-Gaussian (LoG). Feature points are detected by searching local maxima using DoG at various scales of the subject images. The description method extracts a 16×16 neighborhood around each detected feature and further segments the region into sub-blocks, rendering a total of 128 bin values.

4.6.2. ORB Algorithm

The ORB (oriented FAST and rotated BRIEF) algorithm [36] is a combination of FAST (features from accelerated segment test) [73] and BRIEF (binary robust independent elementary features) algorithms [74,75], with some changes to improve accuracy. Figure 10 illustrates the workflow chart of ORB.

ORB parameters are the number of features, scaleFactor, number of octaves level, edgeThreshold, firstLevel, WTA_K and patchSize, and FastThreshold. The max number of keypoints has been set to 5000. The scale factor increased from 1.2 to 1.5. As a result, keypoints and good matches decreased significantly, while total matches decreased slightly. Moreover, it was observed that a scale factor above 1.5 gives reduced results but a slightly higher percentage of better matches than the default value of 1.2. The next test involved the modification of edgeThreshold and sizepatches. From 31, which is the default value, it has been reduced to 28, and at the same time it has been slightly increased and the scalefactor price has increased from 1.2 to 1.5. As a result, keypoints, total matches, and good matches were reduced by half as compared to the previous modifications, while if the scale factor remains at 1.2 and edgeThreshold and size patches will be reduced then, the same number of keypoints is produced in pairs of images; however, we have low total matches and good matches. Then we increased the scale factor to 1.5 and decreased edgeThreshold to 29 and FASTThreshold from 20 to 8. The results were dramatically reduced in both the number of keypoints and matches. Furthermore, we discovered that it makes no sense to increase or decrease edgeThreshold and sizepatch because good matches are increasing; nevertheless, important points on the images are ignored. Finally, FASTThreshold does not offer any significant information or improvement of results. Finally, the ratio was modified to 0.85.

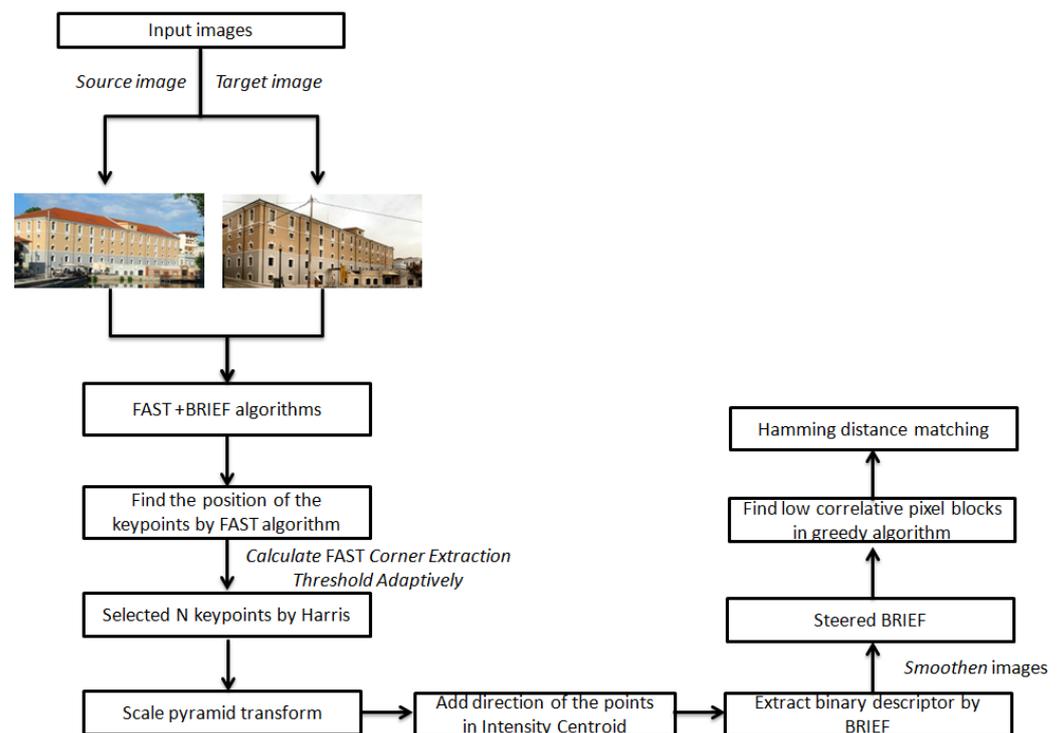


Figure 10. Workflow chart of ORB. The ORB detector takes the intensity threshold between the center pixel and those in a circular ring about the center, and FAST uses a simple measure of corner orientation, the intensity centroid that assumes that a corner's intensity is offset from its center, and this vector is used to refer to an orientation.

4.6.3. BRISK Algorithm

The BRISK (binary robust invariant scalable keypoints) algorithm [37,76] was developed in 2011 as a free alternative to SIFT. It is a robust salient point detection, description, and matching algorithm. The BRISK algorithm utilizes a 16-point FAST detector to identify potential salient points in octaves and intra-octaves of scale-space pyramid. The FAST detector also calculates FAST score S , which is the threshold that still considers an image point as a corner. Then, a non-maxima suppression routine is applied to detect salient points. Figure 11 shows the workflow of BRISK. To identify a test point in the scale-space pyramid as a salient point, the point should satisfy the following conditions:

1. At least nine consecutive pixels within the 16-pixel circle centered on the test point must be sufficiently brighter or darker than the test point;
2. The test point needs to fulfill the maximum condition with respect to its eight neighboring FAST scores S in the same octave;
3. The score of the test point must be larger than the scores of corresponding pixels in the above and below layers [77].

It detects and describes features with invariant scaling and rotational variability. BRISK constructs the description of local image attributes through the grayscale relationship of random pairs of points in the neighborhood of the local image and obtains the description of binary attributes. Two types of pairs are used for sampling: short where the distance is below a set threshold $dist_{max}$ and long which has a distance above $dist_{min}$ pairs. Moreover, its descriptor has a predefined sampling pattern as compared to ORB. Long pairs are used for orientation and short pairs are used for calculating the descriptor by comparing intensities and pixels are sampled over concentric rings.

BRISK parameters are threshold, number of octaves, and patternScale. Threshold is used between the center pixel density and the pixels within the circle around this pixel while the patternScale applies this scale to the pattern used for sampling the neighborhood

of a keypoint. Initially, the tests were conducted only on the octaves. If the default value is increased then only the number of keypoints decreased is negligible, whereas if the value decreases by 1 then the number of keypoints and total matches are decreased. However, the good matches and the best matching rate are increased. Then we tried to modify the value of patternScale. A value greater than 1 (double, i.e., 2) led to a decrease in all results, while a half-doubling of the value (i.e., 0.5) increased all results, but the results are almost similar to those when default parameters are used. In case all parameters decrease, an increase in total matches is observed; however, it is virtual, as all other parameters remain at the same levels as the default parameters. On the other hand, if only threshold is increased and the other two parameters are reduced, then the number of keypoints and total matches is reduced and the success rate of best matching is increased twice. Finally, reducing only the threshold increases all results except the percentage of success of the matching remaining at the same levels, while increasing the threshold reduces all results dramatically. Finally, the ratio was also modified to 0.85. In Table 1, we present in brief the SIFT, ORB, and BRISK properties and characteristics.

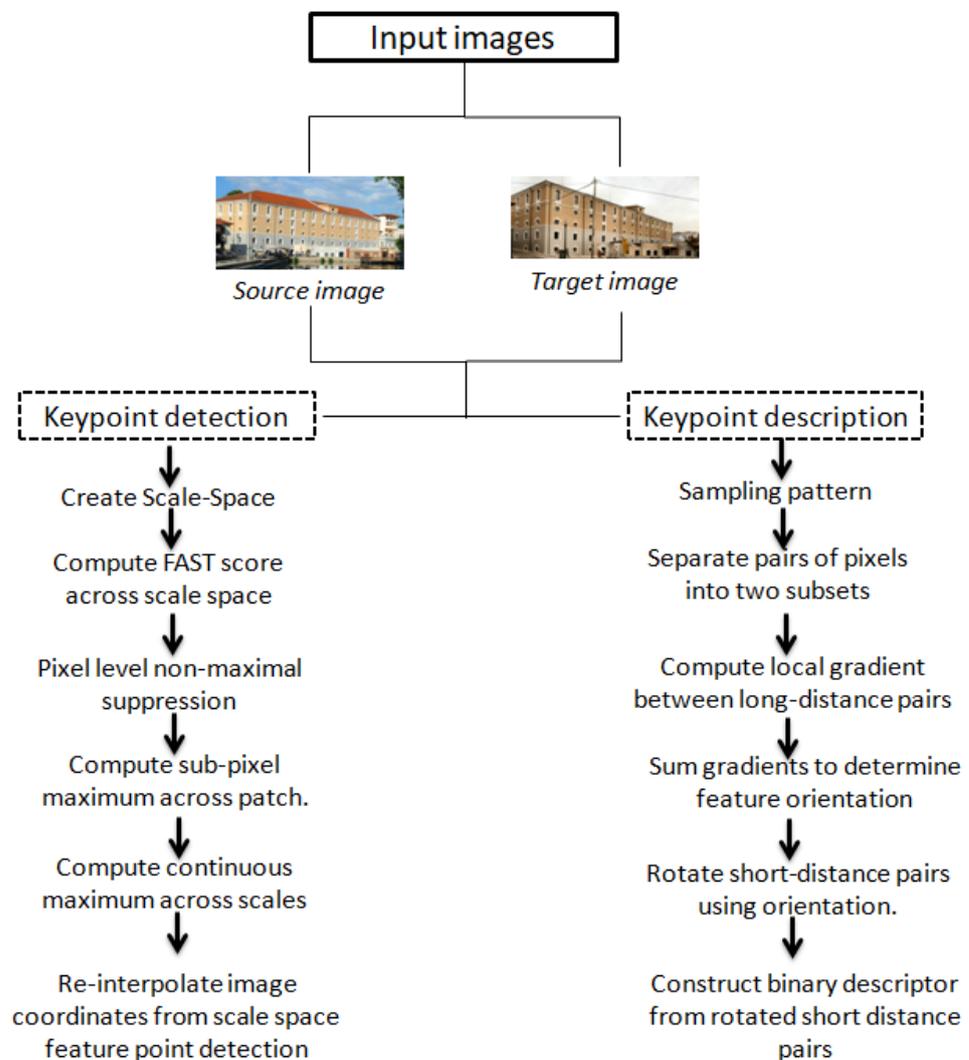
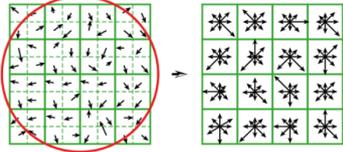
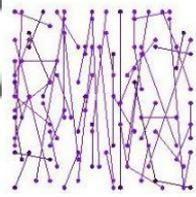
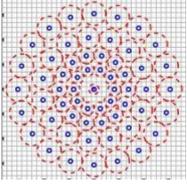


Figure 11. Workflow of BRISK algorithm scale-space keypoint detection: points of interest are identified across both the image and scale dimensions using a saliency criterion. Keypoints are detected in octave layers of the image pyramid. Keypoint description: a sampling pattern consisting of points lying on appropriately scaled concentric circles is applied at the neighborhood of each keypoint to retrieve gray values (processing local intensity gradients and the feature characteristic direction is determined).

Table 1. Illustration of a brief comparison of SIFT, ORB, and BRISK algorithms.

Properties	SIFT (2004)	ORB (2011)	BRISK (2011)
Operators	(a) Detecting keypoints from the multi-scale image space, presented by difference-of-Gaussians (DoG) operator (i.e., approximation of Laplacian-of-Gaussian (LoG)), (b) Keypoint point localization by removing low-contrast and those on edge, (c) assigning orientations to each keypoint based on an orientation histogram, weighted by gradient magnitude and Gaussian-weighted circular window, and (d) providing a unique and robust keypoint descriptor by considering the neighborhood around the keypoint and its orientation histogram [33]	It is a combination of modified FAST (Features from Accelerated Segment Test) which detects corner objects as candidate points, and the Harris Corner score is then utilized to refine them from low-quality points. [73], detection and BRIEF (binary robust independent elementary features) descriptor [74]	It first extracts corners as feature point candidates using the AGAST algorithm [74] and then refines them with the FAST corner score in each scale-space pyramid layer. The illumination robust and rotation invariant descriptor has been generated based on each feature's characteristic direction and simple brightness tests [37]
Keypoints	DoG [57,78]	FAST [79]	AGAST [80]
Detectors type	Blob [81]	Corner [82]	Corner [82]
Descriptors			
Descriptors type and length	Integer vector 128 bytes [35]	Binary string 256 bits [36]	Binary string 64 bytes [37]
Encode information	Gradient-based descriptor [83,84]	Intensity-based descriptor [16]	Intensity-based descriptor [16]
Scale invariant	Yes	Yes (achievable via an image pyramid)	Yes
Rotation invariant	Yes	Yes (achievable via intensity centroid)	Yes
Distance matching	Euclidean [85]	Hamming [86]	Hamming [86]
Constraints	Limited affine changes, high computational cost	Limited affine changes	Limited affine changes, error rate does not exist
Strong points	Robust to illumination fluctuations, noise, Partial occlusion, and minor viewpoint changes in the images [87]	Very Fast, Reduce sensitivity to noise	Robust to noise and affine performance

5. Experimental Results

In this section, we provide a concise and precise description of the experimental results. We itemize the algorithms' parameters and we discuss the keypoints detected and matched, and present our evaluation methods through charts.

5.1. Algorithms Parameters Trial and Error

Table 2 illustrates the default and the modified parameters of algorithms after our trial-and-error tests.

Table 2. Default and modified parameters of algorithms.

Algorithms	Default Parameters	Modified Parameters Trial and Error
SIFT	nfeatures = 0, nOctaveLayers = 3, contrastThreshold = 0.04, edgeThreshold = 10, sigma = 1.6, ratio = 0.7	nfeatures = 0, nOctaveLayers = 4, contrastThreshold = 0.05, edgeThreshold = 8, sigma = 1.5, ratio = 0.85
ORB	nfeatures = 500, scaleFactor = 1.2, nlevels = 8, scoreType = cv.ORB_HARRIS_SCORE, edgeThreshold = 31, firstlevel = 0, scoreType:Harris_score, patchsize = 31, WTA_K = 2, FastThreshold = 20, ratio = 0.7	nfeatures = 5000, scaleFactor = 1.5, nlevels = 8, scoreType = cv.ORB_HARRIS_SCORE, edgeThreshold = 31, firstLevel = 0, WTA_K = 2, patchSize = 31, FastThreshold = 20, ratio = 0.85
BRISK	threshold = 30, octaves = 3, patternScale = 1.0, ratio = 0.7	thresh = 40, octaves = 2, patternScale = 1.0, ratio = 0.85

5.2. Features Detection and Matching

In this subsection, the visualized results from representative images of features extracted and matched by algorithms, both of the same building over time and between different buildings based on certain criteria, are presented. Figure 12 provides some exam-

ples of the tobacco industry from keypoints detected between historic buildings, between historic and modern buildings, and between modern buildings. Keypoints were obtained by SIFT (green), ORB (blue), and BRISK (red) detectors. Figure 13 compares the tobacco industry building with different buildings from various aspects such as date, material, and architectural order.



Figure 12. Tobacco industry building: detection of keypoints. The first pair of buildings is historic (1990–1992), the second pair has 20 years of distance (2000–2021), while the third pair of buildings is modern and of the same year (2022). Green, blue, and red keypoints indicate SIFT, ORB, and BRISK algorithms, respectively.



Figure 13. Keypoints detected by SIFT (green), ORB (blue), and BRISK (red) algorithms between different buildings. The first column includes two modern buildings. The second column includes two stone buildings, and the third column includes two buildings of the same order and at the same time; one is historical and the other modern.

Between historic buildings, the SIFT algorithm detected more keypoints around the point of interest, i.e., windows; however, it did not manage to detect additional features such as walls and vegetation. The ORB algorithm detected more keypoints than SIFT, but they are in scattered form and dense, while the BRISK algorithm detected the least number of keypoints as compared to SIFT and ORB but are more concentrated on corners. Between historic and modern building keypoints were detected by SIFT, mainly on material that divides the texture of the building line under each windows row but not around the interest points such as windows edges and corners. Obviously, the algorithm recognizes the small texture or pattern changes. ORB extracted more and dense keypoints to the entire building, while BRISK detected fewer and sparse keypoints; however, it has more keypoints on all sides of the building. Between modern buildings with different rotation and scales, the SIFT algorithm confirms its advantages as keypoints are detected on all strong interest points; however, it is partially affected by other elements such as water. On the contrary, ORB detected more keypoints, but it seems to present a weakness in the region where the texture or material changes. For instance, there are keypoints denser and scattered near the water. BRISK has the same behavior; however, it has better and more uniform results than ORB.

In Figure 13, the first pair of images includes modern and historic buildings (tobacco industry 2022 and café “Eleutheria” 1940). The second pair contains two buildings with the same structure material (tobacco industry 2016 and Hagia Sofia 2019), while the third pair includes two buildings with almost of the same architecture—however, from different periods (tobacco industry 2022 and elementary school 1990). In the first case, all algorithms provide sparse detections concentrated on corners and at the points of pattern changes. ORB excels related to the density of keypoints, which is higher, while SIFT detects fewer and sparse feature points in the second pair of images. Thus, we notice that SIFT has a weakness to find more features when the texture changes, and we suppose that the spatial range of the feature points is constrained to the buildings with a non-rectangular geometric shape. BRISK detected sparse and scattered keypoints; however, it managed to detect more keypoints on corners than SIFT. Between different buildings by the same material (stone), all algorithms detected keypoints on the interest points; however, ORB and SIFT are affected more by other elements such as vegetation compared to BRISK. In addition, ORB and BRISK detected more dense keypoints in Hagia Sofia compared to SIFT. SIFT recognized better edges and corners on buildings with rectangle regions. In the third case, all the algorithms seem to have the same detection. From this case are seen clearly the advantages of algorithms, despite the diversity of buildings.

Then, we examined the feature matching results from the above buildings and the feature matching of the tobacco industry pairs of images over time (1990–2022). Figure 14 shows the feature matching from tobacco industry pairs. Between historic and modern buildings, SIFT and BRISK algorithms have fewer matches than ORB and confirmed intensively the correspondence problem. In addition, SIFT is also affected by vegetation, while ORB by additional elements such as small walls. Between historic and modern buildings, SIFT and BRISK produced almost the same results; however, the BRISK algorithm presents a weakness, although keypoints extracted were better uniformly distributed, while ORB has more dense matches mainly on the strong interest of points, and it seems to recognize all the textures of buildings. Between modern buildings, the feature matching is more organized and correct, visually. Specifically, SIFT and BRISK managed better performance than ORB and recognized also all the viewing angles of the building.

Figure 15 presents the feature matching results by the tobacco industry building compared to different buildings. The ORB algorithm gave the best results from the similar building shapes aspect while BRISK and SIFT failed to recognize strong points, although of the keypoint extraction phase, they detected them.

Figure 16 presents the feature matching results of the tobacco industry building over time by three algorithms. The building is between 1990–2022. Green, blue, and red colors are matches of the SIFT, ORB, and BRISK algorithms, respectively. SIFT and BRISK

algorithms extract fewer matches both between historic and modern buildings, with an exception. Between modern images where there is no great rotation and scale, the feature matches are dense and concentrated on all buildings. On the contrary, ORB maintains density and uniformity of all pairs of images (i.e., historic and modern) and recognizes new additions and minor changes, such as on walls of historic buildings. Thus, the ORB algorithm is superior to the others, showing a robust visualized performance.

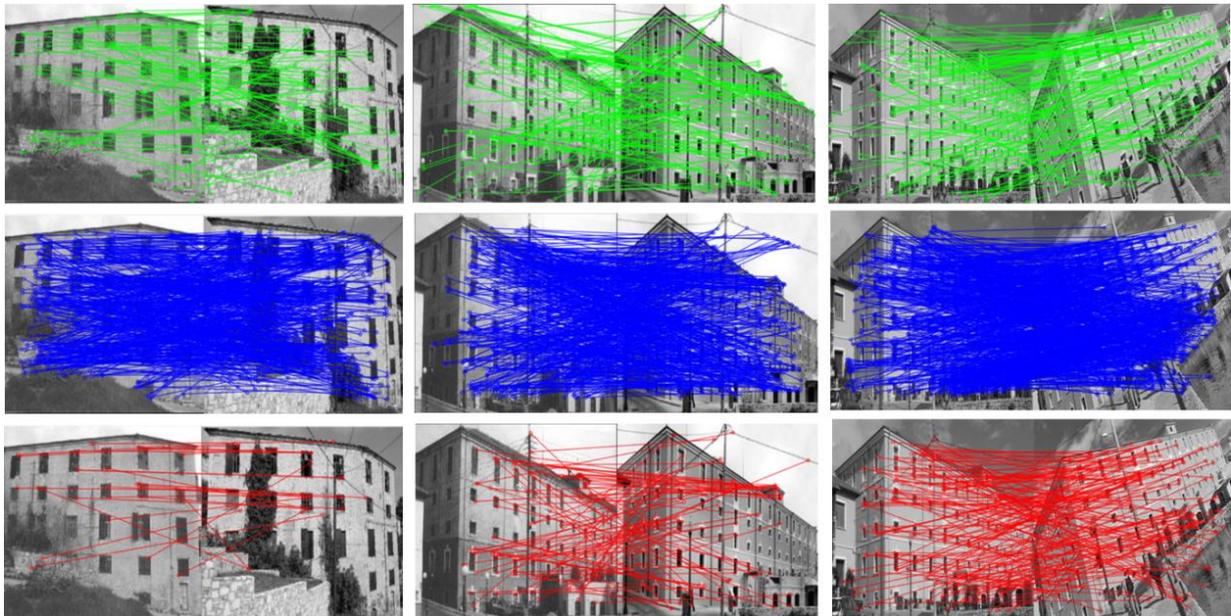


Figure 14. Feature matching. The first column is a pair of historic buildings, the second includes modern and historic buildings, while the third column includes two modern buildings by SIFT (green), ORB (blue), and BRISK (red) by trial-and-error-modified parameters.



Figure 15. Feature matching of different buildings by SIFT, ORB, and BRISK algorithms with modified parameters.

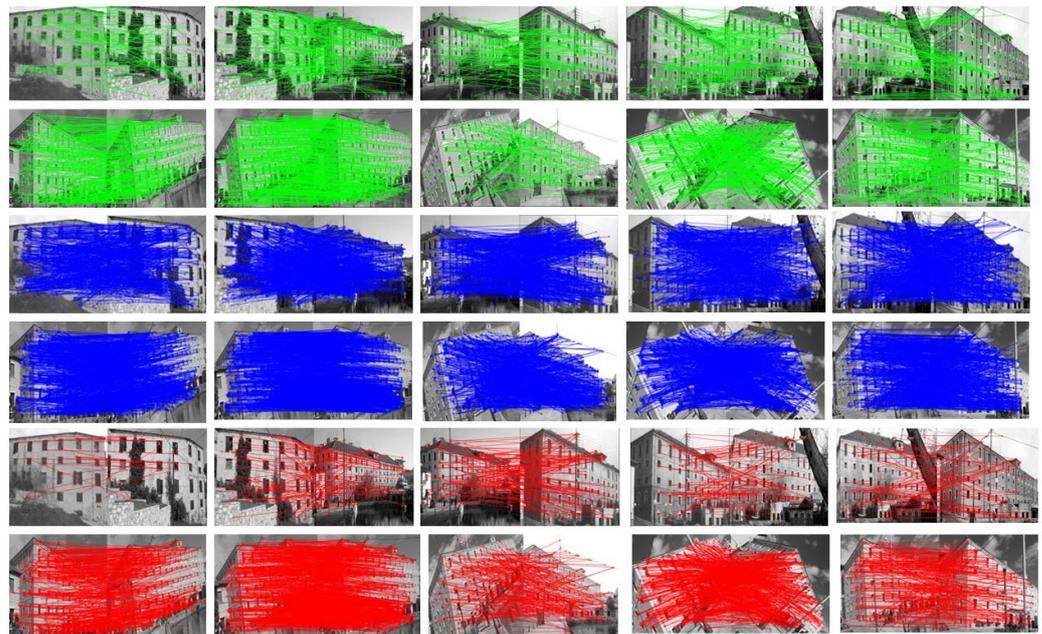


Figure 16. Tobacco industry building. Feature matching historic buildings over time. Green, blue, and red matches are by SIFT, ORB, and BRISK algorithms, respectively.

5.3. Feature Performance Evaluation

In this subsection, the quantified features detection and matching results are presented. Specifically, the number of keypoints detected by the SIFT, ORB, and BRISK algorithms, including the computational cost, are evaluated and compared. Figure 17 shows the total of keypoints from the tobacco industry pairs by algorithms with trial-and-error parameters. The ORB algorithm detects the highest number of features in all pairs of images compared to BRISK, which detects the greater number of features in modern buildings and fewer in historic buildings, while the SIFT algorithm is the opposite of BRISK, i.e., it detected fewer features in modern buildings and more in historic buildings. However, SIFT, ORB, and BRISK detected more keypoints on historic buildings (pair 2), while SIFT and BRISK detected them on modern buildings (pairs 18 and 21) with rotation of the greater number of keypoints, while ORB detected many keypoints on modern buildings without scale and intense rotation as observed in Figure 17.

The next evaluation concerns the evaluation of computational cost, which is an important factor during the feature-matching procedure. The computation efforts of each technique for dealing with modern to historic images can be estimated by recording and analyzing the runtime taken by each technique on each image-pair matching of all test operations. Table 3 shows the keypoints and total matches of the tobacco industry, all pairs (1990–2022), and the runtime results. The BRISK algorithm is the most efficient detector providing the fastest image matching, while SIFT has the greatest computational cost. However, it is normal because it presents also the greatest number of total matches. On the other hand, ORB has less cost compared to SIFT, although the number of total matches is not very different, while the number of total keypoints is almost double.

Table 3. The sum results obtained from pairs of images after applying the three algorithms to datasets, which contain sequences of images for the same building.

Algorithms	Total Keypoints	Total Matches	Runtime (s)
SIFT	76,192	25,246	44.18
ORB	169,233	21,017	31.64
BRISK	101,074	14,881	10.93

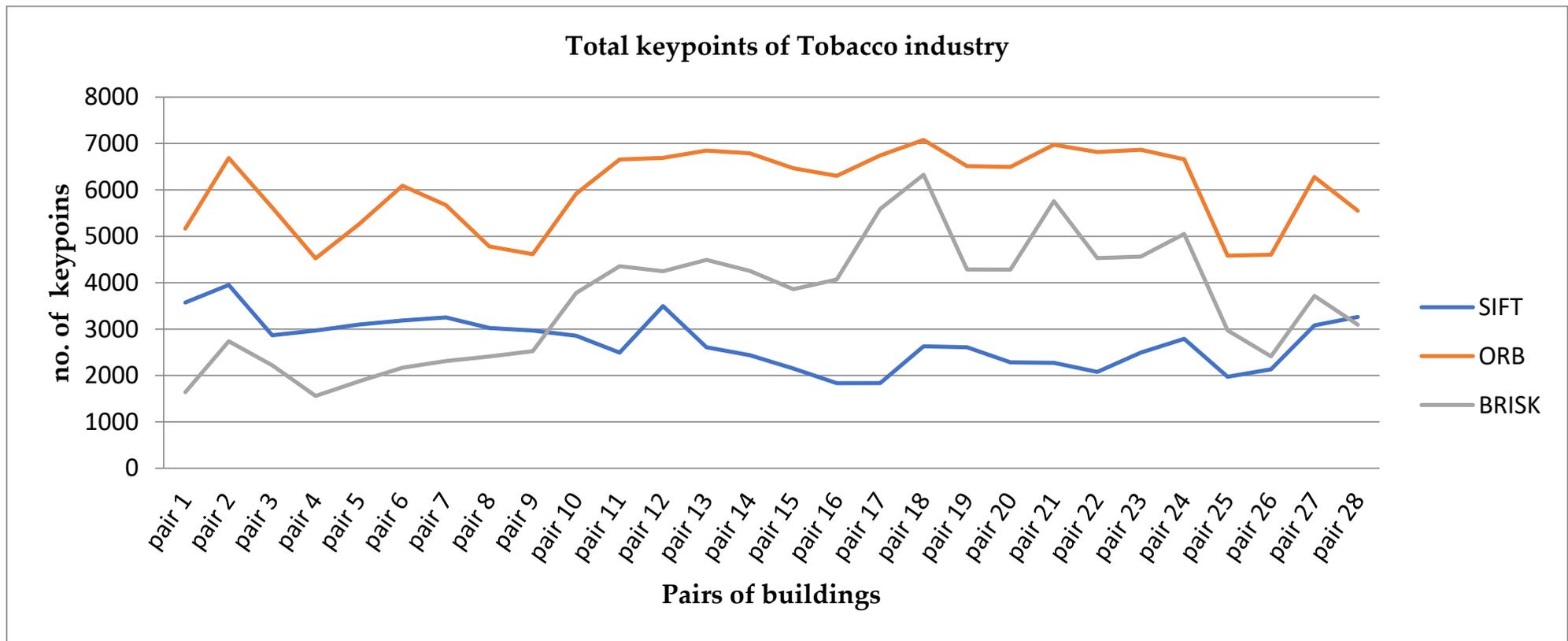


Figure 17. Total keypoints from each pair of images over time using modified parameters of algorithms.

Table 4 shows the number of keypoints detected from each pair of different buildings of the total matches and the computational cost. In this case of different buildings, it is observed that the number of features detected on each pair has significant differences. This is an indication that there are no common points between the images. Nevertheless, BRISK presents less computational cost, while SIFT higher, and ORB comes in second place. It is worth mentioning that the results are also justified based on the levels of human reflection. One may visually conclude whether buildings are similar or not. However, human reflection is also verified by the results. The latter is attributed to the fact that the extracted keypoints in all cases belong to the main structure of the building that does not alter over time. From a detailed observation of the keypoints located in the images, it can be clearly seen that they belong to basic structural elements of the building (windows, definition of floors, etc.) that remain unchanged and not to keypoints on plaster or cracks that are temporary and can be changed over time.

Table 4. Quantitative comparison and computational cost by algorithms for pairs of tobacco industry with other buildings.

Buildings	Algorithms	Keypoints		Total Matches	Runtime (s)
		Image 1	Image 2		
Pair 1	SIFT	1428	900	297	2.00
	ORB	2011	3672	606	2.70
	BRISK	1242	2824	375	0.72
Pair 2	SIFT	1586	1345	320	3.16
	ORB	2514	1978	623	2.10
	BRISK	1282	586	265	0.43
Pair 3	SIFT	2532	1497	177	2.34
	ORB	3424	2462	754	1.93
	BRISK	1329	894	282	0.91
Pair 4	SIFT	1579	2000	372	1.78
	ORB	3096	3527	880	1.54
	BRISK	1126	3382	502	0.75
Pair 5	SIFT	1586	1158	268	1.40
	ORB	2514	3184	674	1.76
	BRISK	1282	2958	492	0.65
Pair 6	SIFT	1042	1010	183	1.12
	ORB	1743	1824	434	1.93
	BRISK	309	549	109	0.42
Pair 7	SIFT	1586	1010	240	1.22
	ORB	2514	1824	553	1.75
	BRISK	1282	549	236	0.85

5.4. Efficient Keypoints Matching

In this subsection, we evaluate the feature matching of the tobacco industry building over time (Figure 18) and the tobacco industry building with different buildings (Figure 19), while then we investigate the effectiveness of keypoints to examine the number of keypoints that were actually used for matching.

The greatest rate of the best matches arose from the pairs with modern buildings that did not have rotation or differences at scale. All the algorithms have similar rates; however, BRISK and ORB present high rates between buildings with intensive scale differences (pairs 21 and 25). On the other hand, BRISK yields the least rates of the best matches mainly to historic buildings, while SIFT follows. Among the pairs, there are also buildings at night (pairs 8 and 9). In this case, the rate of best matches is reduced as compared to the during the day. The low illumination probably is a limiting factor for the satisfactory number of feature detection and matching.

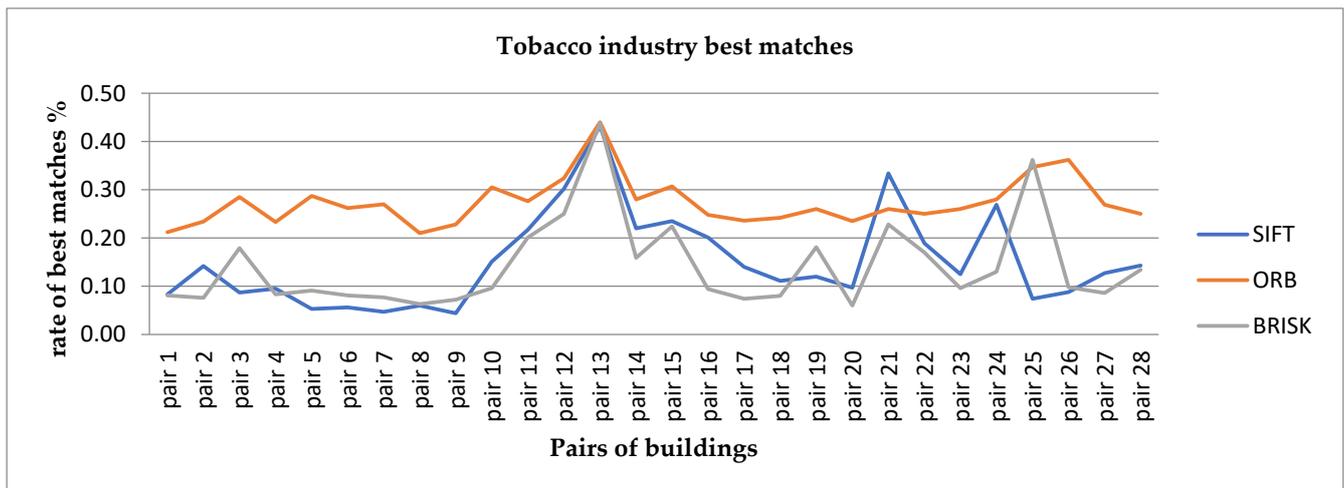


Figure 18. Tobacco industry: rate of the best matches from pairs of buildings over time.

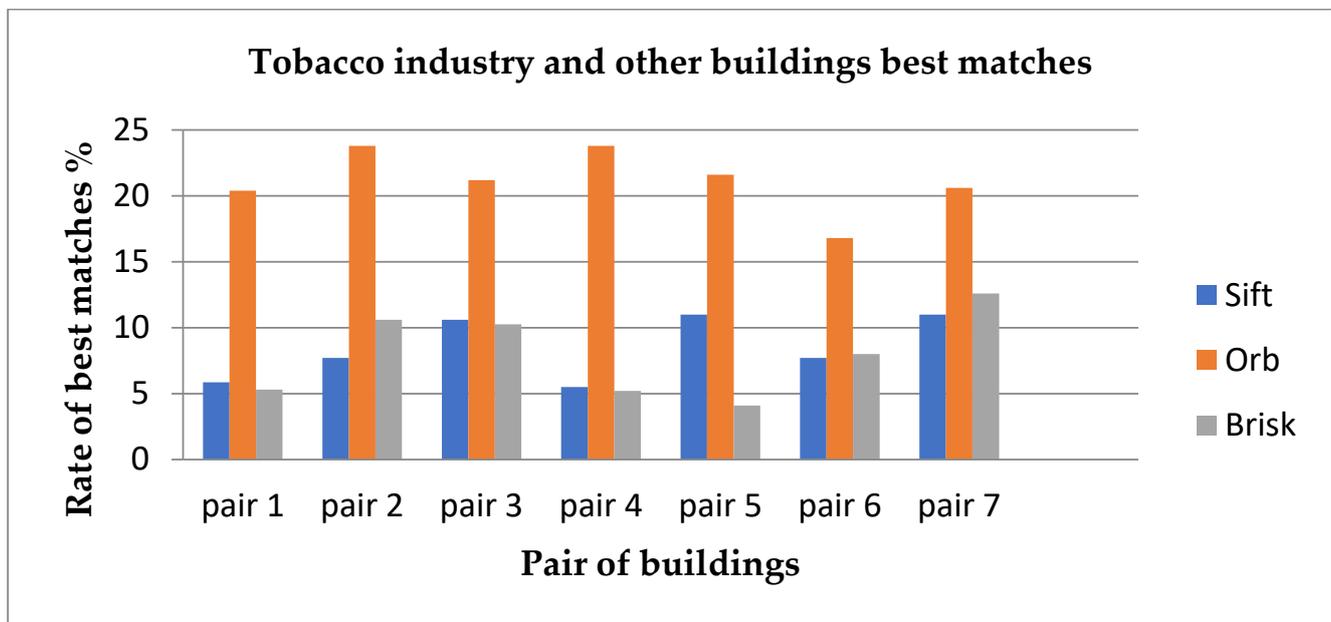


Figure 19. Rate of the best matches of different buildings by algorithms.

According to Figure 19, the ORB algorithm presents the highest rate of best matches while BRISK and SIFT present almost the same results. However, it is noticed that BRISK has the least rate of best matches (pair 5), i.e., between buildings with different architectural styles. (the tobacco industry has a rectangular shape and Hagia Sofia church an irregular shape).

Figure 20 shows the effectiveness of the tobacco industry building over time, while Figure 21 shows the effectiveness between the tobacco industry and other buildings. For the tobacco industry building over time, the BRISK algorithm has the least rate of keypoints matching both historic (1999–2000) and modern (pairs 21 and 25) buildings. On the other hand, the peak rate is presented by all algorithms to modern buildings without rotation (pair 13) and scale or with intensive rotation (pair 21), while BRISK presents the least rate on historic buildings (pair 3) and SIFT on different view-angle modern building (pair 7). Finally, all buildings have a tendency to decrease the rate of the keypoints actually used.

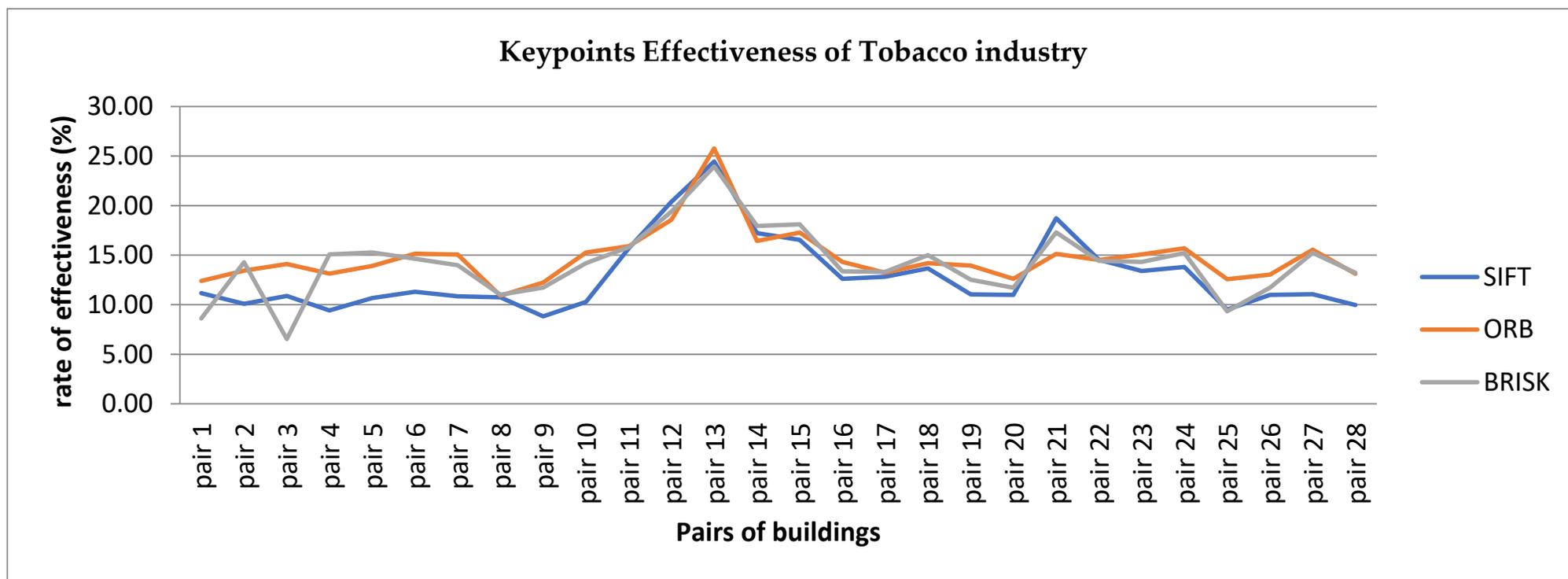


Figure 20. Effectiveness of keypoint matching in pairs of the same building over time.

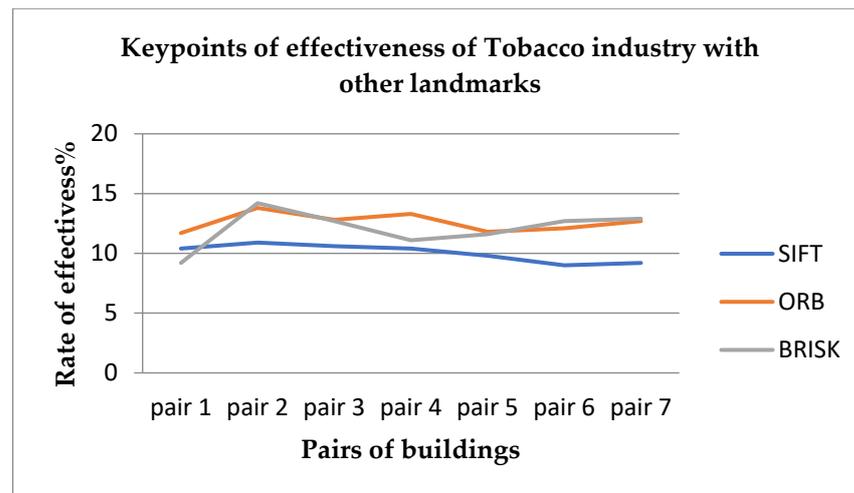


Figure 21. Effectiveness of keypoint matching in pairs of different building over time.

In Figure 21, BRISK and ORB present similar results to all pairs of images, while the rate by SIFT is similar to all pairs of buildings. However, in pair 4, while ORB presents an increase, SIFT presents a decrease. This is justified by the fact that between two buildings with different materials, SIFT fails to find similarities. Nevertheless, all results are virtual because there are not in reality common elements among the buildings.

Image Matching Accuracy

In this subsection, we investigate the final matching accuracy. First, we use an average precision index to evaluate coarsely the ratio of best matches/total matches, while since none of the algorithms completely guarantees correct feature matching results, we evaluate false positive matches from the best matches. The assessment is done visually without the threshold value because in such a case the results will probably not be unbiased. Figure 22 presents the average precision of good matches from the tobacco industry pairs over time. ORB provides the highest precision, contrary to BRISK, while SIFT comes in second place. In general, algorithms provide results from moderate to high precision. This is reasonable because the precision is affected by matching criteria. The stricter the matching criteria are, the greater the number of correct matches and the higher the accuracy.

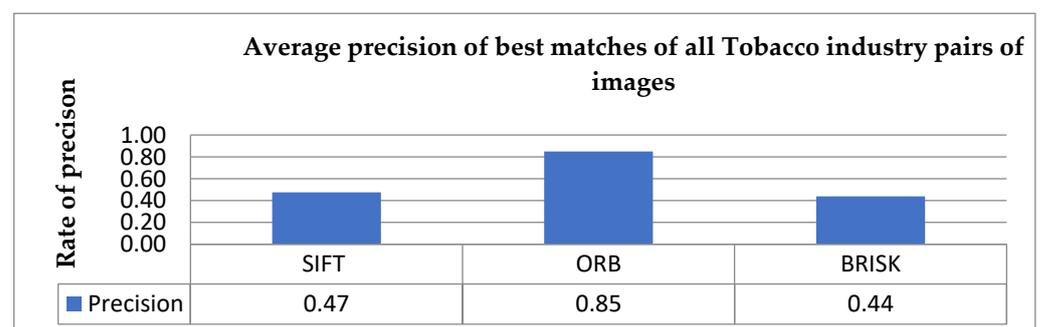


Figure 22. Average precision matching from all pairs of images.

Figure 23 illustrates the average precision of the tobacco industry building from each date with all the others. All algorithms provide similarities to their results. The ORB algorithm is proved to have higher precision of best matches on modern buildings that are under similar conditions and come up to 100% (pair 15). The SIFT algorithm is following; however, it presents rates over 50% for both historic (pair 2) and modern buildings (pairs 10, 16, 24, and 28), i.e., buildings without differences. The BRISK algorithm has the fewer rates; however, on modern buildings precision is similar under all conditions.

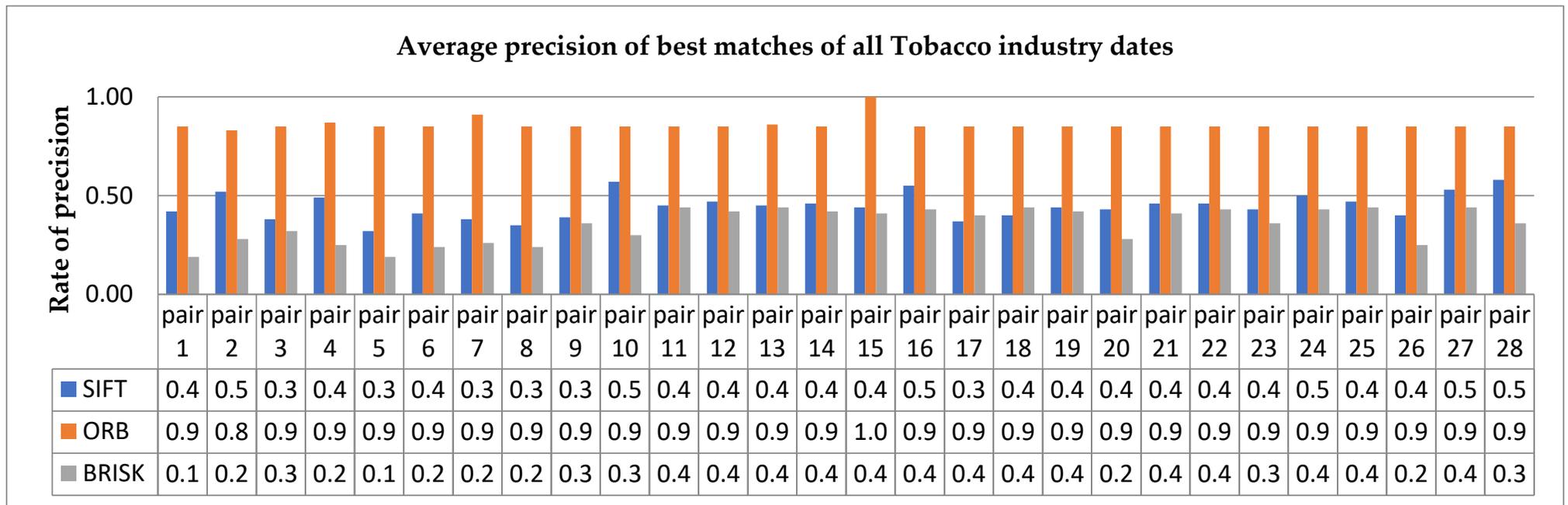


Figure 23. Tobacco industry. Average precision each year compared to all the others.

Figure 24 shows the tobacco industry precision with other landmarks' precision of the best matches. The SIFT algorithm has almost half the precision of BRISK. In particular, SIFT provides lower rates mainly on buildings with similar dates but different architectural styles. The ORB algorithm has similar results, while BRISK presents less precision to modern and different architectural-style buildings.

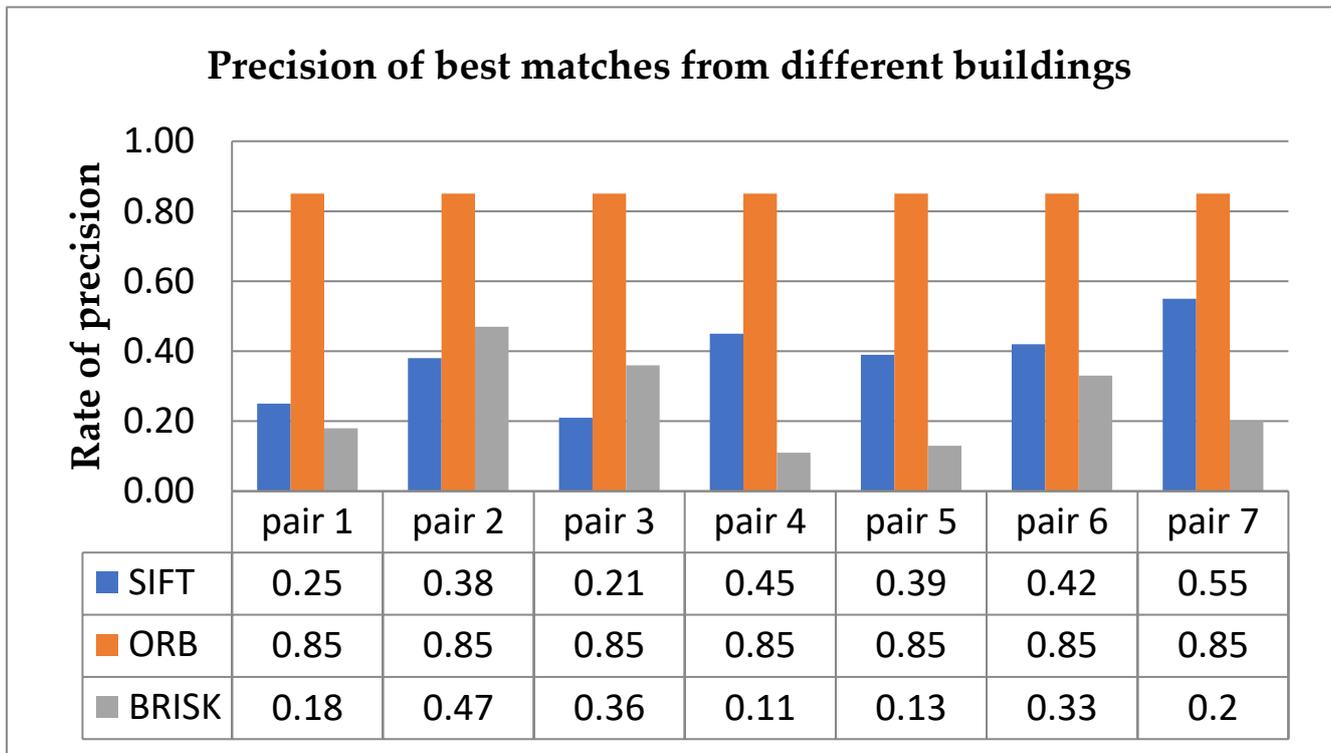


Figure 24. Rate of precision among pairs of different buildings.

Finally, the last evaluation is the false positive of incorrectly detected feature matching. Table 5 shows pairs of the tobacco industry building over time results. According to the results on the buildings over a 15-year time (pair 1–pair 5) the correct matches fail at 100% by all algorithms except ORB, which rises near half. The reason for the failure is partly justified because the viewpoint of images is completely different, i.e., images (pair 1) illustrate only the facade of the building while the texture remains the same. In images where the building is captured only from the facade, all algorithms except SIFT succeed at half rates compared to the previous. In general, between modern buildings, the rate of false positives ranges in low percentages by all algorithms with some exceptions. For instance, the SIFT and BRISK algorithms have high rates of mismatching (pair 10). This pair of buildings has different illumination. In addition, SIFT, ORB, and BRISK appear with equally high rates of false matching in buildings with steep rotation and different scales (pairs 19, 20, 25, and 27). Finally, in fact, the evaluation of false positive matches is not undertaken among different landmarks because there are no common features.

Table 5. Rates (%) of false positives of feature matching.

Buildings	SIFT	ORB	BRISK
Pair 1	97.67	51.47	68.00
Pair 2	100.00	58.58	100.00
Pair 3	93.55	74.74	93.00
Pair 4	100.00	96.63	100.00

Table 5. Cont.

Buildings	SIFT	ORB	BRISK
Pair 5	100.00	97.75	97.06
Pair 6	98.88	99.23	92.90
Pair 7	94.74	82.05	26.14
Pair 8	100.00	91.63	100.00
Pair 9	100.00	99.17	100.00
Pair 9	84.90	55.41	86.99
Pair 11	21.80	42.16	33.86
Pair 12	7.57	20.95	44.14
Pair 13	10.18	18.21	37.59
Pair 14	15.48	10.77	35.01
Pair 15	24.90	10.33	51.67
Pair 16	47.44	52.35	77.30
Pair 17	71.56	40.37	79.19
Pair 18	53.39	11.48	39.09
Pair 19	62.90	44.23	38.36
Pair 20	68.32	67.82	94.52
Pair 21	25.51	12.96	34.48
Pair 22	27.69	8.57	24.28
Pair 23	32.06	39.25	70.62
Pair 24	23.62	13.85	64.47
Pair 25	100.00	85.89	98.71
Pair 26	94.12	71.76	92.06
Pair 27	41.45	60.92	96.08
Pair 28	90.86	95.95	90.85

6. Discussion—Conclusions

In this paper, we applied feature extraction and matching techniques to pairs of images, using SIFT, ORB, and BRISK detectors and descriptors to identify historic buildings over time. Comparing historic and modern buildings, the SIFT algorithm detected fewer numbers of features on modern buildings and higher numbers on historic buildings with rotations contrary to BRISK, which detected more features on modern buildings, also without rotation. In addition, although SIFT detected keypoints to window corners, small texture and pattern changes are not recognized, while the ORB algorithm detected more features on modern buildings without scale and rotation changes and are not concentrated on strong points such as corners, compared to the BRISK algorithm. This means that ORB also equally takes into account other elements in the scene.

On the feature-matching level, SIFT presented fewer rates of best matches to historical buildings after BRISK, as compared to ORB, which performed the highest rate of best matching. It is notable that it recognizes all the viewpoints of buildings. Moreover, ORB and BRISK had also a high rate of matching between buildings with various scales.

On the basis of the computational cost, BRISK was more efficient, providing fast matching to all cases, while SIFT had the greatest computational cost but also the greatest number of total matches. ORB had a high computational cost—however, fewer than SIFT. BRISK might be the correct choice in simpler cases that require speed.

The actual number of keypoints was used for feature matching. SIFT provided fewer rates, in particular to modern buildings with different view angles, while ORB had similar results to all pairs of images and BRISK the least. Image-matching precision results showed that from greater to less, ORB presented the best performance and BRISK and SIFT followed, while comparing each date to all others, ORB retained first place, whereas SIFT and BRISK followed, mainly on historical buildings. The same results were also found among pairs of different buildings. Finally, on the basis of false positives, SIFT and BRISK presented higher rates than BRISK, which had half.

According to the above results, SIFT performs the best on modern buildings under rotations, while ORB under scale changes. In addition, SIFT and BRISK had higher accuracy for image rotation changes. On the other hand, ORB moves stable and high detection rates on match tests for feature extraction and matching. Besides, it presents high rates of isolated, aged buildings. In general, ORB is more robust and reliable than SIFT, although it is less scale-invariant, while BRISK seems to be competitively related to SIFT as it performs under rotation and scale changes.

Before algorithm characteristics, we took into account some constraints that were unavoidable. For instance, some old images have been converted into digital format. This had an impact on the loss of basic and valuable information. In addition, since some data are derived from books, the quality of paper degradation is also an important factor.

The buildings in the urban scene have various complex patterns, structural materials, and different shapes. In addition, in urban environments exist other elements, such as buildings, vegetation, or water, and these lead to the appearance of shadows or hidden parts of study buildings. On the other hand, the different conditions of captured buildings make the identification of buildings more difficult. However, our algorithms in many cases found homologous points on the same material. The experimental results confirmed that our methods, up to a point, are capable of identifying the minor changes in buildings over time as the feature matches are more sparse or dense and the feature detection caused by the various factors are not always on the strong points such as (corners and edges). Finally, it is notable that although SIFT is considered the best for many applications, our experiments showed that the ORB and BRISK algorithms are also efficient. However, these standard methods are not sufficient but are satisfactory for identifying landmarks over time.

Future work will consider the computational optimization of the algorithms' parameters to examine the trade-off between enhanced performance and processing time. An extended dataset is also undergoing, with the aim to further investigate the performance of data-intensive techniques, such as deep learning, to the problem under study. Although this study replied to the initial research questions, new challenges are emerging. Deep learning methods to learn dense feature representations gain more and more ground. In the future, it is expected that modern methods such as photogrammetry and remote sensing will contribute to a better understanding of scenery, including urban historic buildings, using dense point clouds and orthophotomaps that provide further metric information, as only from geometry or texture it is difficult to extract conclusions.

Author Contributions: Conceptualization, K.A.T. and G.A.P.; methodology, K.A.T. and G.A.P.; software, K.A.T.; validation, G.A.P., S.C. and G.E.T.; formal analysis, K.A.T.; investigation, K.A.T.; resources, K.A.T.; data curation, K.A.T.; writing—original draft preparation, K.A.T.; writing—review and editing, E.V., S.C. and G.E.T.; visualization, K.A.T. and S.C.; supervision, G.A.P.; project administration, G.A.P., S.C. and G.E.T.; funding acquisition, G.A.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: This work was supported by the MPhil program “Advanced Technologies in Informatics and Computers”, hosted by the Department of Computer Science, International Hellenic University, Kavala, Greece.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Whitehead, A.; Opp, J. Timescapes: Putting History in Your Hip Pocket. In Proceedings of the Computers and Their Applications Conference CATA, Honolulu, HI, USA, 4–6 March 2013; pp. 261–266.
2. Kabir, S.R.; Akhtaruzzaman, M.; Haque, R. Performance Analysis of Different Feature Detection Techniques for Modern and Old Buildings. In Proceedings of the 3rd International Conference on Recent Trends and Applications in Computer Science and Information Technology, Tiranë, Albania, 23 November 2018; pp. 120–127.
3. Rebec, K.M.; Deanovič, B.; Oostwegel, L. Old Buildings Need New Ideas: Holistic Integration of Conservation-Restoration Process Data Using Heritage Building Information Modelling. *J. Cult. Herit.* **2022**, *55*, 30–42. [\[CrossRef\]](#)
4. Mahinda, M.C.P.; Udhyani, H.P.A.J.; Alahakoon, P.M.K.; Kumara, W.G.C.W.; Hinas, M.N.A.; Thamboo, J.A. Development of An Effective 3D Mapping Technique for Heritage Structures. In Proceedings of the 2021 3rd International Conference on Electrical Engineering (EECon), Colombo, Sri Lanka, 24 September 2021; pp. 92–99. [\[CrossRef\]](#)
5. Tuytelaars, T.; Mikolajczyk, K. Local Invariant Feature Detectors: A Survey. *FNT Comput. Graph. Vis.* **2007**, *3*, 177–280. [\[CrossRef\]](#)
6. Santosh, D.; Achar, S.; Jawahar, C.V. Autonomous Image-Based Exploration for Mobile Robot Navigation. In Proceedings of the 2008 IEEE International Conference on Robotics and Automation, Pasadena, CA, USA, 19–23 May 2008; pp. 2717–2722. [\[CrossRef\]](#)
7. Milford, M.; McKinnon, D.; Warren, M.; Wyeth, G. Feature-based Visual Odometry and Featureless Place Recognition for SLAM in 2.5 D Environments. In Proceedings of the Australasian Conference on Robotics and Automation (ACRA 2011), Melbourne Australia, 7–9 December 2011; pp. 1–8.
8. Sminchisescu, C.; Bo, L.; Ionescu, C.; Kanaujia, A. Feature-Based Pose Estimation. In *Visual Analysis of Humans*; Moeslund, T.B., Hilton, A., Krüger, V., Sigal, L., Eds.; Springer: London, UK, 2011; pp. 225–251. [\[CrossRef\]](#)
9. Hu, Y. Research on a Three-Dimensional Reconstruction Method Based on the Feature Matching Algorithm of a Scale-Invariant Feature Transform. *Math. Comput. Model.* **2011**, *54*, 919–923. [\[CrossRef\]](#)
10. Nixon, M.S.; Aguado, A.S. *Feature Extraction and Image Processing*, 1st ed.; Newnes: Oxford, UK; Boston, MA, USA, 2002.
11. Amiri, M.; Rabiee, H.R. RASIM: A Novel Rotation and Scale Invariant Matching of Local Image Interest Points. *IEEE Trans. Image Process.* **2011**, *20*, 3580–3591. [\[CrossRef\]](#)
12. Weng, D.; Wang, Y.; Gong, M.; Tao, D.; Wei, H.; Huang, D. DERF: Distinctive Efficient Robust Features From the Biological Modeling of the P Ganglion Cells. *IEEE Trans. Image Process.* **2015**, *24*, 2287–2302. [\[CrossRef\]](#)
13. Levine, M.D. Feature Extraction: A Survey. *Proc. IEEE* **1969**, *57*, 1391–1407. [\[CrossRef\]](#)
14. Ha, Y.-S.; Lee, J.; Kim, Y.-T. Performance Evaluation of Feature Matching Techniques for Detecting Reinforced Soil Retaining Wall Displacement. *Remote Sens.* **2022**, *14*, 1697. [\[CrossRef\]](#)
15. Viola, P.; Wells III, W.M. Alignment by maximization of mutual information. *Int. J. Comput. Vis.* **1997**, *24*, 137–154. [\[CrossRef\]](#)
16. Myronenko, A.; Song, X. Intensity-Based Image Registration by Minimizing Residual Complexity. *IEEE Trans. Med. Imaging* **2010**, *29*, 1882–1891. [\[CrossRef\]](#)
17. Liu, X.; Tian, Z.; Ding, M. A Novel Adaptive Weights Proximity Matrix for Image Registration Based on R-SIFT. *AEU-Int. J. Electron. Commun.* **2011**, *65*, 1040–1049. [\[CrossRef\]](#)
18. Leng, C.; Xiao, J.; Li, M.; Zhang, H. Robust Adaptive Principal Component Analysis Based on Intergraph Matrix for Medical Image Registration. *Comput. Intell. Neurosci.* **2015**, *2015*, 829528. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Friedrichs, K.; Münster, S.; Kröber, C.; Bruschke, J. Creating Suitable Tools for Art and Architectural Research with Historic Media Repositories. In *Digital Research and Education in Architectural Heritage*; Münster, S., Friedrichs, K., Niebling, F., Seidel-Grzesińska, A., Eds.; Communications in Computer and Information Science; Springer International Publishing: Cham, Switzerland, 2018; Volume 817, pp. 117–138. [\[CrossRef\]](#)
20. Ali, H.; Whitehead, A. Subset Selection for Landmark Modern and Historic Images. In Proceedings of the 2nd International Conference on Signal and Image Processing, Geneva, Switzerland, 21–22 March 2015; pp. 69–79. [\[CrossRef\]](#)
21. Ali Heider, K.; Whitehead, A. Modern to Historic Image Matching: ORB/SURF an Effective Matching Technique. In Proceedings of the Computers and Their Applications, Las Vegas, NV, USA, 24–26 March 2014.
22. Becker, A.-K.; Vornberger, O. Evaluation of Feature Detectors, Descriptors and Match Filtering Approaches for Historic Repeat Photography. In *Image Analysis*; Felsberg, M., Forssén, P.-E., Sintorn, I.-M., Unger, J., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2019; Volume 11482, pp. 374–386. [\[CrossRef\]](#)
23. Anderson-Bell, J.; Schillaci, C.; Lipani, A. Predicting non-residential building fire risk using geospatial information and convolutional neural networks. *Remote Sens. Appl. Soc. Environ.* **2021**, *21*, 100470. [\[CrossRef\]](#)
24. Agarwal, S.; Snavely, N.; Simon, I.; Seitz, S.M.; Szeliski, R. Building Rome in a Day. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 72–79. [\[CrossRef\]](#)
25. Uttama, P.L.; Delalandre, M.; Ogier, J.M. Segmentation and Retrieval of Ancient Graphic Documents. In *Graphics Recognition. Ten Years Review and Future Perspectives*; Springer: Cham, Switzerland, 2006; pp. 88–98.

26. Ali, H.; Whitehead, A. Feature Matching for Aligning Historical and Modern Images. *Int. J. Comput. Appl.* **2014**, *21*, 188–201.
27. Wolfe, R. Modern to Historical Image Feature Matching. 2015. Available online: <http://robbiewolfe.ca/programming/honoursproject/report.pdf> (accessed on 3 February 2023).
28. Wu, G.; Wang, Z.; Li, J.; Yu, Z.; Qiao, B. Contour-Based Historical Building Image Matching. In Proceedings of the 2nd International Symposium on Image Computing and Digital Medicine—ISICDM, Chengdu, China, 13–15 October 2018; ACM Press: Chengdu, China, 2018; pp. 32–36. [[CrossRef](#)]
29. Hasan, M.S.; Ali, M.; Rahman, M.; Arju, H.A.; Alam, M.M.; Uddin, M.S.; Allayear, S.M. Heritage Building Era Detection Using CNN. *IOP Conf. Ser. Mater. Sci. Eng.* **2019**, *617*, 012016. [[CrossRef](#)]
30. Maiwald, F.; Schneider, D.; Henze, F.; Münster, S.; Niebling, F. Feature matching of historical images based on geometry of quadrilaterals. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* **2018**, *XLII-2*, 643–650. [[CrossRef](#)]
31. Yue, L.; Li, H.; Zheng, X. Distorted Building Image Matching with Automatic Viewpoint Rectification and Fusion. *Sensors* **2019**, *19*, 5205. [[CrossRef](#)] [[PubMed](#)]
32. Si, L.; Hu, X.; Liu, B. Image Matching Algorithm Based on the Pattern Recognition Genetic Algorithm. *Comput. Intell. Neurosci.* **2022**, *2022*, 7760437. [[CrossRef](#)]
33. Edward, J.; Yang, G.-Z. RANSAC with 2D Geometric Cliques for Image Retrieval and Place Recognition. In Proceedings of the CVPR Workshop, Boston, MA, USA, 7–12 June 2015.
34. Avrithis, Y.; Toliás, G. Hough Pyramid Matching: Speeded-Up Geometry Re-Ranking for Large Scale Image Retrieval. *Int. J. Comput. Vis.* **2014**, *107*, 1–19. [[CrossRef](#)]
35. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [[CrossRef](#)]
36. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An Efficient Alternative to SIFT or SURF. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2564–2571. [[CrossRef](#)]
37. Leutenegger, S.; Chli, M.; Siegwart, R.Y. BRISK: Binary Robust Invariant Scalable Keypoints. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2548–2555. [[CrossRef](#)]
38. Smith, S.M.; Brady, J.M. Susan—a new approach to low level image processing. *Int. J. Comput. Vis.* **1997**, *23*, 45–78. [[CrossRef](#)]
39. Nixon, M.; Aguado, A. *Feature Extraction & Image Processing for Computer Vision*; Elsevier: Amsterdam, The Netherlands, 2012. [[CrossRef](#)]
40. Tsafirir, D.; Tsafirir, I.; Ein-Dor, L.; Zuk, O.; Notterman, D.A.; Domany, E. Sorting Points into Neighborhoods (SPIN): Data Analysis and Visualization by Ordering Distance Matrices. *Bioinformatics* **2005**, *21*, 2301–2308. [[CrossRef](#)]
41. Harris, C.; Stephens, M. A Combined Corner and Edge Detector. In *Proceedings of the Alvey Vision Conference 1988*; Alvey Vision Club: Manchester, UK, 1988; pp. 23.1–23.6. [[CrossRef](#)]
42. Shi, F.; Huang, X.; Duan, Y. Robust Harris-Laplace Detector by Scale Multiplication. In *Advances in Visual Computing*; Bebis, G., Boyle, R., Parvin, B., Koracin, D., Kuno, Y., Wang, J., Wang, J., Wang, J., Pajarola, R., Lindstrom, P., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5875, pp. 265–274. [[CrossRef](#)]
43. Sarangi, S.; Sahidullah, M.; Saha, G. Optimization of Data-Driven Filterbank for Automatic Speaker Verification. *Digit. Signal Process.* **2020**, *104*, 102795. [[CrossRef](#)]
44. Mutlag, W.K.; Ali, S.K.; Aydam, Z.M.; Taher, B.H. Feature Extraction Methods: A Review. *J. Phys. Conf. Ser.* **2020**, *1591*, 012028. [[CrossRef](#)]
45. Kumar, G.; Bhatia, P.K. A Detailed Review of Feature Extraction in Image Processing Systems. In Proceedings of the 2014 Fourth International Conference on Advanced Computing & Communication Technologies, Rohtak, India, 8–9 February 2014; pp. 5–12. [[CrossRef](#)]
46. Wang, X.; Jabri, A.; Efros, A.A. Learning Correspondence from the Cycle-Consistency of Time. *Comput. Vis. Pattern Recognit.* **2019**, 2566–2576. [[CrossRef](#)]
47. Muhammad, U.; Tanvir, M.; Khurshid, K. Feature Based Correspondence: A Comparative Study on Image Matching Algorithms. *Int. J. Adv. Comput. Sci. Appl.* **2016**, *7*. [[CrossRef](#)]
48. Zhao, C.; Cao, Z.; Yang, J.; Xian, K.; Li, X. Image Feature Correspondence Selection: A Comparative Study and a New Contribution. *IEEE Trans. Image Process.* **2020**, *29*, 3506–3519. [[CrossRef](#)]
49. Howe, P.D.; Livingstone, M.S. Binocular Vision and the Correspondence Problem. *J. Vis.* **2005**, *5*, 800. [[CrossRef](#)]
50. Torresani, L.; Kolmogorov, V.; Rother, C. Feature Correspondence Via Graph Matching: Models and Global Optimization. In *Computer Vision—ECCV 2008*; Forsyth, D., Torr, P., Zisserman, A., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2008; Volume 5303, pp. 596–609. [[CrossRef](#)]
51. Kolmogorov, V.; Zabih, R. Computing Visual Correspondence with Occlusions Using Graph Cuts. In Proceedings of the Eighth IEEE International Conference on Computer Vision. ICCV 2001, Vancouver, BC, Canada, 7–14 July 2001; Volume 2, pp. 508–515. [[CrossRef](#)]
52. Kabbai, L.; Abdellaoui, M.; Douik, A. Image Classification by Combining Local and Global Features. *Vis. Comput.* **2019**, *35*, 679–693. [[CrossRef](#)]
53. Mikolajczyk, K.; Mikolajczyk, K. Scale & Affine Invariant Interest Point Detectors. *Int. J. Comput. Vis.* **2004**, *60*, 63–86. [[CrossRef](#)]
54. Keyvanpour, M.R.; Vahidian, S.; Ramezani, M. HMR-Vid: A Comparative Analytical Survey on Human Motion Recognition in Video Data. *Multimed. Tools Appl.* **2020**, *79*, 31819–31863. [[CrossRef](#)]

55. Chen, L.; Rottensteiner, F.; Heipke, C. Feature Detection and Description for Image Matching: From Hand-Crafted Design to Deep Learning. *Geo-Spat. Inf. Sci.* **2021**, *24*, 58–74. [[CrossRef](#)]
56. Krig, S. Interest Point Detector and Feature Descriptor Survey. In *Computer Vision Metrics*; Springer International Publishing: Cham, Switzerland, 2016; pp. 187–246. [[CrossRef](#)]
57. Hassaballah, M.; Abdelmgeid, A.A.; Alshazly, H.A. Image Features Detection, Description and Matching. In *Image Feature Detectors and Descriptors*; Awad, A.I., Hassaballah, M., Eds.; Studies in Computational Intelligence; Springer International Publishing: Cham, Switzerland, 2016; Volume 630, pp. 11–45. [[CrossRef](#)]
58. Leng, C.; Zhang, H.; Li, B.; Cai, G.; Pei, Z.; He, L. Local Feature Descriptor for Image Matching: A Survey. *IEEE Access* **2019**, *7*, 6424–6434. [[CrossRef](#)]
59. González-Aguilera, D.; Ruiz de Oña, E.; López-Fernandez, L.; Farella, E.; Stathopoulou, E.K.; Toschi, I.; Remondino, F.; Rodríguez-Gonzálvez, P.; Hernández-López, D.; Fusiello, A.; et al. Photomatch: An open-source multi-view and multi-modal feature matching tool for photogrammetric applications. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* **2020**, *XLIII-B5-2020*, 213–219. [[CrossRef](#)]
60. Sun, J.; Shen, Z.; Wang, Y.; Bao, H.; Zhou, X. LoFTR: Detector-Free Local Feature Matching with Transformers. *Comput. Vis. Pattern Recognit.* **2021**, 8922–8931. [[CrossRef](#)]
61. Zitová, B.; Flusser, J. Image Registration Methods: A Survey. *Image and Vision Computing* **2003**, *21*, 977–1000. [[CrossRef](#)]
62. Litjens, G.; Kooi, T.; Bejnordi, B.E.; Setio, A.A.A.; Ciompi, F.; Ghafoorian, M.; van der Laak, J.A.W.M.; van Ginneken, B.; Sánchez, C.I. A Survey on Deep Learning in Medical Image Analysis. *Med. Image Anal.* **2017**, *42*, 60–88. [[CrossRef](#)]
63. Flusser, J.; Suk, T. A Moment-Based Approach to Registration of Images with Affine Geometric Distortion. *IEEE Trans. Geosci. Remote Sens.* **1994**, *32*, 382–387. [[CrossRef](#)]
64. Goshtasby, A.; Stockman, G.; Page, C. A Region-Based Approach to Digital Image Registration with Subpixel Accuracy. *IEEE Trans. Geosci. Remote Sens.* **1986**, *GE-24*, 390–399. [[CrossRef](#)]
65. Hsieh, Y.C.; McKeown, D.M.; Perlant, F.P. Performance Evaluation of Scene Registration and Stereo Matching for Cartographic Feature Extraction. *IEEE Trans. Pattern Anal. Machine Intell.* **1992**, *14*, 214–238. [[CrossRef](#)]
66. Hellier, P.; Barillot, C. Coupling Dense and Landmark-Based Approaches for Nonrigid Registration. *IEEE Trans. Med. Imaging* **2003**, *22*, 217–227. [[CrossRef](#)]
67. Mikolajczyk, K.; Schmid, C. A Performance Evaluation of Local Descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1615–1630. [[CrossRef](#)] [[PubMed](#)]
68. Noble, F.K. Comparison of OpenCV’s Feature Detectors and Feature Matchers. In Proceedings of the 2016 23rd International Conference on Mechatronics and Machine Vision in Practice (M2VIP), Nanjing, China, 28–30 November 2016; pp. 1–6. [[CrossRef](#)]
69. Dhana Lakshmi, M.; Mirunalini, P.; Priyadharsini, R.; Mirnalinee, T.T. Review of Feature Extraction and Matching Methods for Drone Image Stitching. In *Proceedings of the International Conference on ISMAC in Computational Vision and Bio-Engineering 2018 (ISMAC-CVB)*, Palladam, India, 16–17 May 2018; Pandian, D., Fernando, X., Baig, Z., Shi, F., Eds.; Lecture Notes in Computational Vision and Biomechanics; Springer International Publishing: Cham, Switzerland, 2019; Volume 30, pp. 595–602. [[CrossRef](#)]
70. Spasova, V.G. Experimental evaluation of keypoints detector and descriptor algorithms for indoors person localization. *Annu. J. Electron.* **2014**, *8*, 85–87.
71. Vijayan, V.; Kp, P. FLANN Based Matching with SIFT Descriptors for Drowsy Features Extraction. In Proceedings of the 2019 Fifth International Conference on Image Information Processing (ICIIP), Shimla, India, 15–17 November 2019; pp. 600–605. [[CrossRef](#)]
72. Luo, Z.; Zhou, L.; Bai, X.; Chen, H.; Zhang, J.; Yao, Y.; Li, S.; Fang, T.; Quan, L. ASLFeat: Learning Local Features of Accurate Shape and Localization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020. [[CrossRef](#)]
73. Rosten, E.; Drummond, T. Machine Learning for High-Speed Corner Detection. In *Computer Vision—ECCV 2006*; Leonardis, A., Bischof, H., Pinz, A., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2006; Volume 3951, pp. 430–443. [[CrossRef](#)]
74. Calonder, M.; Lepetit, V.; Strecha, C.; Fua, P. BRIEF: Binary Robust Independent Elementary Features. In *Computer Vision—ECCV 2010*; Daniilidis, K., Maragos, P., Paragios, N., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2010; Volume 6314, pp. 778–792. [[CrossRef](#)]
75. Martin, K.A.C. A BRIEF History of the “Feature Detector”. *Cereb Cortex* **1994**, *4*, 1–7. [[CrossRef](#)] [[PubMed](#)]
76. Tareen, S.A.K.; Saleem, Z. A Comparative Analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK. In Proceedings of the 2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), Sukkur, Pakistan, 3–4 March 2018; pp. 1–10. [[CrossRef](#)]
77. Azimi, E.; Behrad, A.; Ghaznavi-Ghouschi, M.B.; Shanbehzadeh, J. A Fully Pipelined and Parallel Hardware Architecture for Real-Time BRISK Salient Point Extraction. *J. Real-Time Image Proc.* **2019**, *16*, 1859–1879. [[CrossRef](#)]
78. Awad, A.I.; Hassaballah, M. (Eds.) Image Feature Detectors and Descriptors: Foundations and Applications. In *Studies in Computational Intelligence*; Springer International Publishing: Cham, Switzerland, 2016; Volume 630. [[CrossRef](#)]
79. Chen, J.; Shan, S.; He, C.; Zhao, G.; Pietikäinen, M.; Chen, X.; Gao, W. WLD: A Robust Local Image Descriptor. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 1705–1720. [[CrossRef](#)]
80. Zhang, H.; Wohlfeil, J.; Griebßbach, D. Extension and evaluation of the AGAST feature detector. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.* **2016**, *III-4*, 133–137. [[CrossRef](#)]

81. Xiong, X.; Choi, B.-J. Comparative Analysis of Detection Algorithms for Corner and Blob Features in Image Processing. *Int. J. Fuzzy Log. Intell. Syst.* **2013**, *13*, 284–290. [[CrossRef](#)]
82. Ghafoor, A.; Iqbal, R.N.; Khan, S. Robust Image Matching Algorithm. In Proceedings of the 4th EURASIP Conference focused on Video/Image Processing and Multimedia Communications (IEEE Cat. No.03EX667), Zagreb, Croatia, 2–5 July 2003; Volume 1, pp. 155–160. [[CrossRef](#)]
83. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-Based Learning Applied to Document Recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [[CrossRef](#)]
84. Dalal, N.; Triggs, B. Histograms of Oriented Gradients for Human Detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 886–893. [[CrossRef](#)]
85. Jakubovic, A.; Velagic, J. Image Feature Matching and Object Detection Using Brute-Force Matchers. In Proceedings of the 2018 International Symposium ELMAR, Zadar, Croatia, 16–19 September 2018; pp. 83–86. [[CrossRef](#)]
86. Norouzi, M.; Fleet, D.J.; Salakhutdinov, R.R. Hamming distance metric learning. In Proceedings of the Neural Information Processing Systems (NeurIPS 2012), Lake Tahoe, NV, USA, 3–8 December 2012; Volume 25, pp. 1061–1069.
87. Lu, Y.; Liu, A.-A.; Su, Y.-T. Detection in Biomedical Images. In *Computer Vision for Microscopy Image Analysis*; Elsevier: Amsterdam, The Netherlands, 2021; pp. 131–157. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.