Article

Image-Based Delineation and Classification of Built Heritage Masonry

Noelia Oses [1], Fadi Dornaika [2,3,]* and Abdelmalik Moujahid [2]

[1] Fundación Zain Fundazioa, Vitoria-Gasteiz 01006, Spain; E-Mail: noelia@nofcon.com

[2] Department of Computer Science and Artificial Intelligence, University of the Basque Country UPV/EHU, San Sebastian 20018, Spain; E-Mail: jibmomoa@gmail.com

[3] IKERBASQUE, Basque Foundation for Science, Bilbao 48011, Spain

* Author to whom correspondence should be addressed; E-Mail: fadi.dornaika@ehu.es; Tel.: +34-943-010-834.

Abstract: Fundación Zain is developing new built heritage assessment protocols. The goal is to objectivize and standardize the analysis and decision process that leads to determining the degree of protection of built heritage in the Basque Country. The ultimate step in this objectivization and standardization effort will be the development of an information and communication technology (ICT) tool for the assessment of built heritage. This paper presents the ground work carried out to make this tool possible: the automatic, image-based delineation of stone masonry. This is a necessary first step in the development of the tool, as the built heritage that will be assessed consists of stone masonry construction, and many of the features analyzed can be characterized according to the geometry and arrangement of the stones. Much of the assessment is carried out through visual inspection. Thus, this process will be automated by applying image processing on digital images of the elements under inspection. The principal contribution of this paper is the automatic delineation the framework proposed. The other contribution is the performance evaluation of this delineation as the input to a classifier for a geometrically characterized feature of a built heritage object. The element chosen to perform this evaluation is the stone arrangement of masonry walls. The validity of the proposed framework is assessed on real images of masonry walls.

1. Introduction

In the last decade, machine vision techniques have been increasingly used in order to assist the process of cultural heritage documentation, preservation and restoration [1–3]. Image-based approaches, for example, have been used to automatically detect built heritage decay [4–8]. 3D digital modeling for cultural heritage has recently received a lot of attention from the scientific community [9–14]. Much work has been done with the objective of automating the process of 3D modeling of cultural heritage sites or buildings [15–17]. However, the semantic categorization (categorization based on high-level, overall meaning of the scene as opposed to low-level, individual details) of the related scenes (images or 3D models) has received little attention. 3D modeling of cultural heritage is useful for visualization, digital archiving and sharing for education, research and conservation. Thus, machine vision and photogrammetry techniques become appealing in order to achieve the related objectives. In the field of geometrical documentation of heritage, research has been carried out in order to automate the processing of 3D point clouds. Rodríguez *et al*. [18] present a semi-automatic method to draw straight lines in the 3D point cloud. This method requires human interaction to specify the segments of interest in the 2D sections. Some low-cost computer tools for documenting and analyzing built heritage have already been proposed [19–22].

Fundación Zain is developing new built heritage assessment protocols that standardize the protection evaluation process in the Basque Country and, also, working on a built heritage analysis and classification ICT tool that implements said protocols. The remainder of the paper (this paper is an extended version of the conference paper, [23]) is organized as follows. Section 2 describes the problem statement and the contributions of this paper. Section 3 presents some related, state-of-the-art work. Section 4 describes the proposed automatic stone masonry delineation framework. Section 5 provides the results of the delineation and an evaluation of its performance as the input to a classifier for a geometrically characterized feature. Finally, Section 6 provides conclusions and future work.

2. Problem Statement: Automatic Image-Based Delineation and Classification of Masonry

2.1. General Context

The objective of the built heritage protection ICT tool under development is to assist and to speed up the protection process, that is, the analysis and decision process that leads to determining the degree of protection of built heritage. Oses and Azkarate [24] provide a more in-depth description of the tool and its context. Much of the assessment in this process is carried out through visual inspection. Thus, it will be automated by applying image processing on digital images of the elements under inspection. The use of digital images will contribute to making the protection process automatic and sustainable over time. Many of the features analyzed can be characterized geometrically, and often, this characterization is related to the arrangement of the stone masonry blocks. This paper presents the ground work carried out to make this tool possible: the automatic delineation of the masonry. Delineation, in the context of our work, refers to the extraction of the outline of the construction blocks.

The new protection process will have a protocol for each different type of built heritage (stone bridges, rural traditional houses, defensive buildings, *etc*.). Each protocol will analyze many different elements

of the object under analysis. The validity of this delineation will be shown by evaluating its performance for one of the elements assessed in the protocols: the arrangement of the masonry.

This paper makes two main contributions. First, it proposes a new framework for the extraction of geometric primitives from a given masonry image. Second, based on the extracted geometric features, it provides a study about the statistical feature extraction and selection for the purpose of the automatic categorization of masonry. We stress that the objective of the paper is not to propose a novel processing algorithm; rather, we are interested in developing a framework that combines existing image processing and machine learning tools for solving a challenging, fine-grained categorization problem, namely, the image-based classification of masonry walls.

Our proposed framework for the automatic delineation and classification of built heritage has two phases. In the first phase, the test image undergoes a series of processing steps in order to extract a set of straight line segments from which a statistical signature is inferred. This set of straight line segments constitutes a partial delineation of blocks. In the second phase, the statistical signature is filtered out and classified using classifiers. The proposed framework is outlined in Figure 1.

Figure 1. The flowchart of the proposed delineation (Section 4) and classification (Section 5.2) framework.
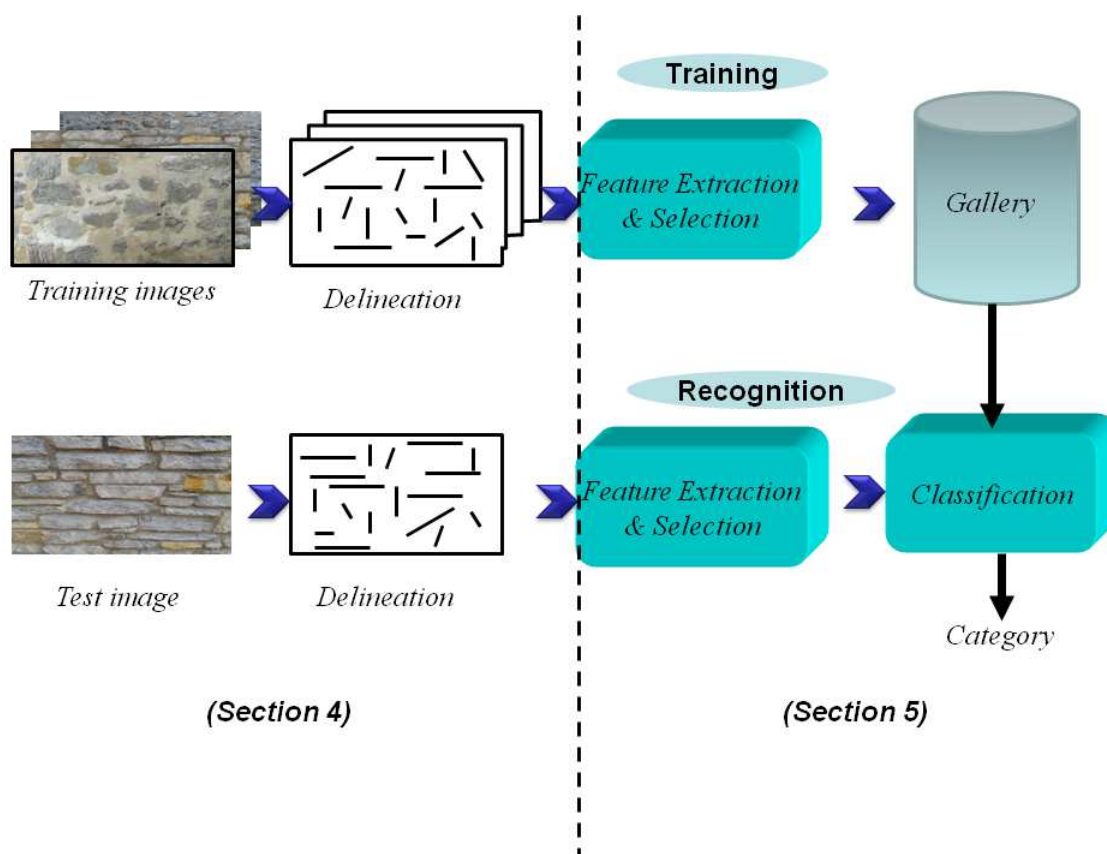

2.2. Automatic Delineation of Masonry: Challenges

The built heritage that will be assessed using Fundación Zain's protection protocols consists of stone masonry construction. Our objective is to develop image-based tools that help automate the application

of these protocols. In this paper, we address the image-based extraction of the masonry arrangement for automatic classification. The fully automatic, image-based delineation of individual stones of built heritage can be very challenging, due to the characteristics of the objects to be delineated and the environment in which they are. Stone masonry walls of built heritage are built with many types of stones and mortar. Often, the same wall contains stones of different colors, sizes, shapes and materials. This can be appreciated looking at the various walls shown in the different figures in this paper. These walls belong to construction that is old and outdoors, thus having suffered from prolonged weathering, leading to different degradation patterns, such as discoloration and biological colonization [25]. Therefore, unlike crisp (*i.e.*, precisely delimited) and structured regions that are usually present in modern building scenes, the blocks in the walls under consideration do not have structured and crisp boundaries. Biological colonization, *i.e.*, vegetation growing in the wall, results in occlusions. As a consequence, the physical boundaries that delineate individual stones, when imaged, do not always offer meaningful information about the real boundaries of the stones. These degradation patterns will, often, have a negative impact on the performance of the delineation algorithms.

Furthermore, the image capturing is performed in an uncontrolled environment, with lighting conditions possibly changing for different captures. Images can, therefore, have bright and dark areas and shadows depending on the sun's position. In the case of bridge walls, the images can also contain reflections of the water. These undesired effects in the images generate intensity gradients that are, often, completely unrelated to the physical delineation of the masonry, and as a consequence, conventional edge-based methods are not useful, since the signal-to-noise ratio is very low.
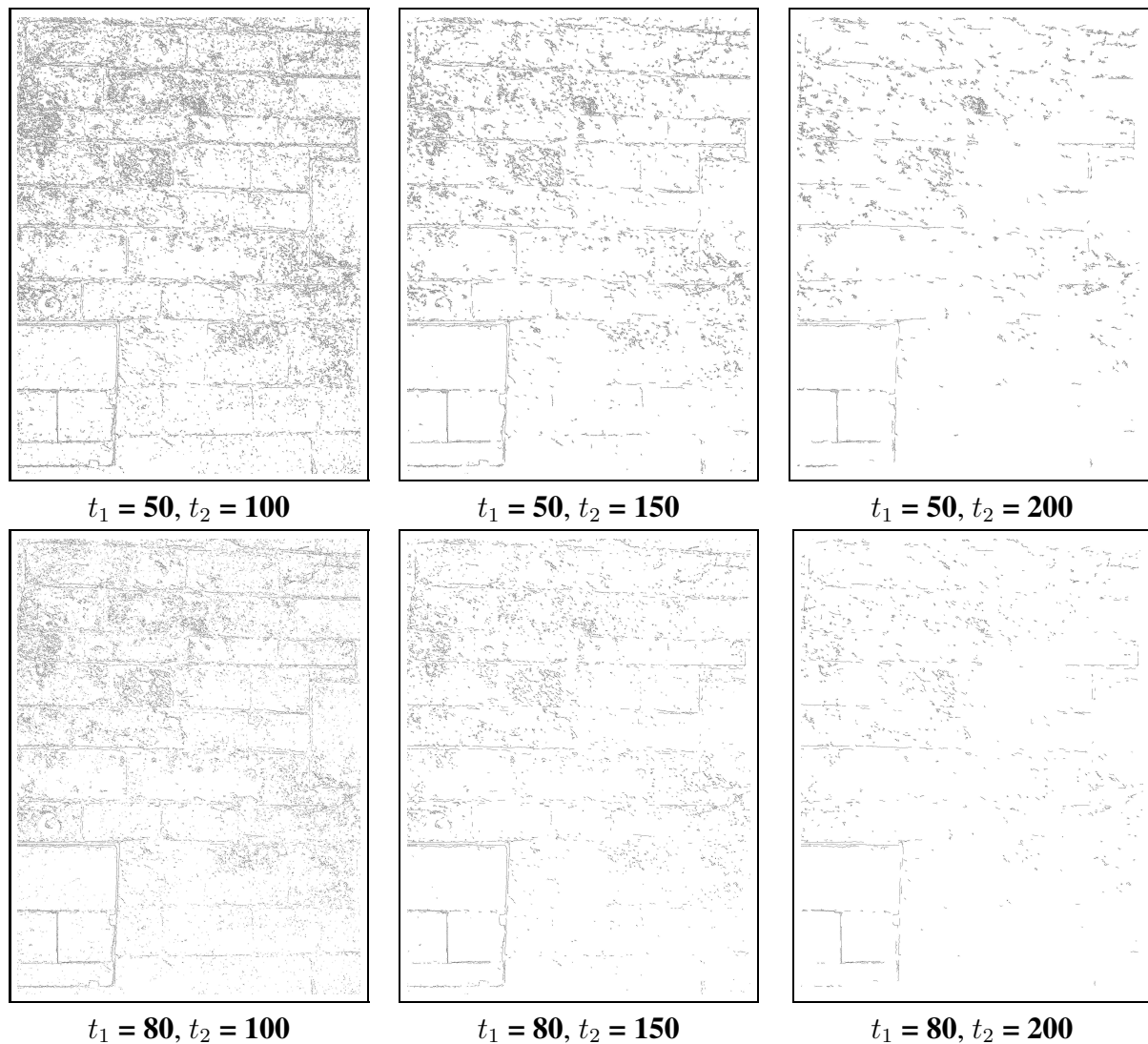
Figure 2. The original image (scaled) that will be used as a running example to illustrate the delineation method proposed.
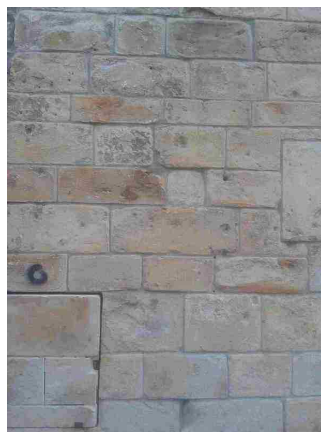
At first glance, one may think of applying automatic image-based granulometry techniques in order to delineate the individual stones and, then, infer the category from the geometric description of the delineated individual blocks. However, using classic gradient map based approaches does not help much in obtaining a good delineation. Very often, gradient map methods applied to our images result in noise. If we try to adjust the parameters of the gradient map generation algorithm to reduce the amount of noise, then we lose valuable information. Figure 2 shows the original image (scaled) that will be used as a running example to illustrate the delineation method proposed. Figure 3 shows the result of applying the Canny edge detection algorithm [26] to the original image. This sequence of images shows that, with

the Canny algorithm, we get too much noise or not enough edge points to obtain a meaningful (partial) delineation for many stones. These challenges motivate the development of a new delineation method (Section 4) that does not rely on conventional edge detection methods.

Figure 3. Images showing the result of applying the Canny edge detector to the original image with an aperture size of three and different threshold values. t_1 denotes the lower threshold and t_2 the upper threshold.

$t_1 = 50, t_2 = 100$ $t_1 = 50, t_2 = 150$ $t_1 = 50, t_2 = 200$

$t_1 = 80, t_2 = 100$ $t_1 = 80, t_2 = 150$ $t_1 = 80, t_2 = 200$

2.3. Built Heritage Element Classification

As stated earlier, the protocols will analyze many different features of each built heritage object type before reaching a conclusion. One of the features that can be characterized geometrically and, thus, is a good candidate to be analyzed automatically through digital image processing is the type of masonry arrangement. This is the feature we have chosen to prove that the information obtained with the automatic delineation framework proposed in this paper is meaningful and can be used successfully in the automatic classification of this type of feature.

Three classes of masonry arrangements have been defined in these protocols: the first class are blocks (usually irregular) not arranged in rows (Figure 4a); the second are irregular blocks arranged in rows (Figure 4b); and, third, regular (rectangular) blocks arranged in rows (Figure 4c). The masonry walls used in this experiment are located at different sites in the Basque Country: Durango and Urdaibai. The dataset contains 86 wall images; 33 belong to Class 1, 15 belong to Class 2 and the remaining 38 to Class 3.

Figure 4. Examples of the three categories to be discriminated. (**a**) Class 1; (**b**) Class 2; (**c**) Class 3.

(**a**)   (**b**)   (**c**)
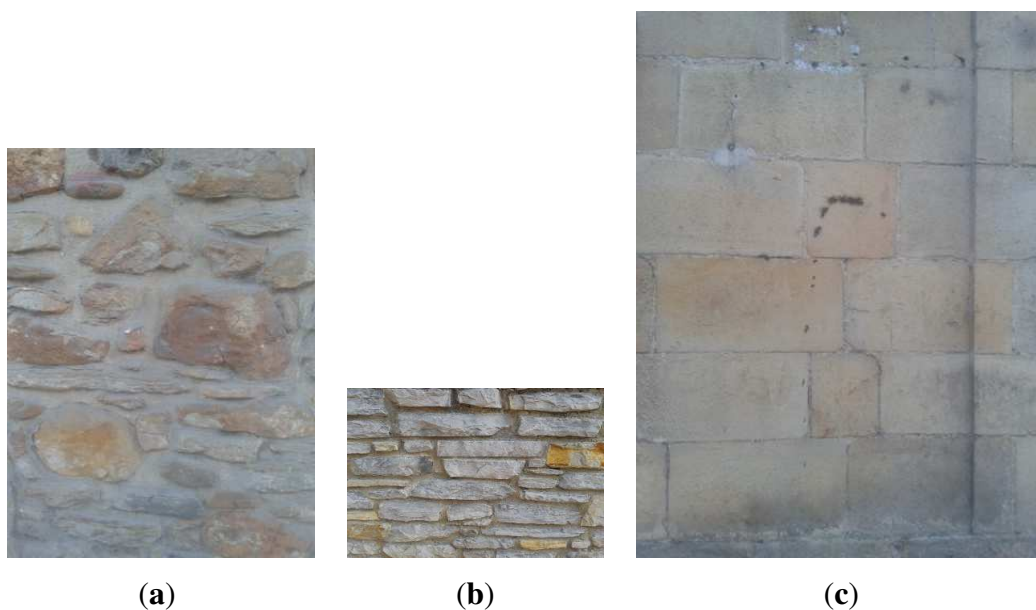
At the beginning of the project, the decision was made to use inexpensive and easily-operated tools, such as a standard digital camera. The digital camera used for image acquisition is a Pentax Optio M30, a 7.1-megapixel budget compact camera. When acquiring the image of a wall, the camera was held in a fronto-parallel configuration, so the perspective effect was reduced. Furthermore, the distance between the camera and the wall was roughly kept constant for all walls (at about 1.5 m). The size of the original images was not fixed. However, the captured images should contain a large portion of the wall, so that good statistics can be collected for that wall.

We stress the fact that this classification problem is quite different from the classic scene categorization problem [27], where each class can refer to a different concept (sea, street, building, car, human). Traditional image classification tasks are aimed at classifying objects with large inter-class differences in semantics. Oppositely, fine-grained visual categorization (FGVC) is an emerging topic in computer vision [28,29]. A fine-grained image collection typically contains many categories sharing similar semantics (bird categories or human faces, for example). In many FGVC problems, it is even difficult for a human to recognize all the categories. Obviously, our masonry classification problem belongs to the family of FGVC, as can be appreciated from the illustrative class images.

The recent emergence of local binary patterns (LBP) has also led to significant progress in applying texture methods to various computer vision problems [30,31]. These approaches are able to discriminate image textures (e.g., face recognition). However, they are prone to failure for correctly classifying our

masonry data, because the masonry arrangement categories that we need to classify are fully defined by the geometric arrangement and shape of stones. Thus, texture discrimination tools cannot be used in our case, since the samples of the same masonry category can have many types of textures, and even the same instance (a wall) may contain several different textures. This classification problem is discussed in-depth in Section 5.2.

3. Existing Approaches for Automatic Delineation

3.1. General Purpose Image Processing Tools

Currently, there are software packages that offer functionality related to the delineation of digital images. The goal of these packages is to vectorize a bitmap image of general content. Many of them bear the word "trace" in their name, which can be understood as the vectorization of a raster image. Vector graphics, as opposed to raster images, represent an image by using geometric objects, such as polygons and curves. The vectorization of the image is, precisely, the objective of the automatic delineation discussed in this paper. However, the objective of the existing software packages for the vectorization of images is to make these scalable and, thus, be able to correctly view them at different resolutions. On the other hand, the ultimate objective of the automatic delineation of this paper is to precisely delimit the contours of the different elements of built heritage, so that it can be used to perform metric analysis on these.

For instance, AutoTrace (http://autotrace.sourceforge.net/) and Potrace (http://potrace.sourceforge.net/) are, probably, the most popular ones, due to their open source and freeware nature. Potrace interprets black and white bitmaps to produce a set of curves, which makes up the vectorization. AutoTrace's objective is to offer functionality similar to that of Corel PowerTRACE or Adobe Streamline (now replaced by the Live Trace functionality of Adobe Illustrator CS2). The latter two, together with Xara Designer (Bitmap Tracer), are the most well known commercial packages for image editing.

Figure 5. Potrace output for the original image (Figure 2).

Inkscape (http://www.inkscape.org/en/) is an open source vector graphics editor that uses Potrace to obtain delineations. It uses three different filters to convert raw images (color or gray scale) into black and white images, so that they can be fed to Potrace. These filters are the following. The "Brightness

Threshold" filter merely uses the sum of the red, green and blue (or shades of gray) of a pixel as an indicator of whether it should be considered black or white. The "Optimal Edge Detection" filter uses the edge detection algorithm devised by J. Canny as a way of quickly finding isoclines of similar contrast. The "Color Quantization" filter will find edges where colors change by obtaining a color segmentation first and, then, deciding black/white on whether the color has an even or odd index. Figure 5 shows the result of applying Potrace to our original image (Figure 2). The vectorization is based on color segmentation and edge detection based on gradient maps. Thus, it is not delimiting the contour of the stones, but delimiting the contours of colors and gradient changes. This, clearly, is not the delineation we need.

3.2. Image-Based Granulometry

In the last two decades, image analysis for rock particles has become a hot topic of research, and a number of image systems (discussed next) have been developed for segmenting and measuring rock particles in different applications, such as gravitational flows, conveyor belts, rock piles and laboratories. Some of them are still under development [32]. Since the main goal is to measure the size of particles using their images, these systems include a delineation module that attempts to segment the particles in the images in order to infer information about sizes and their distribution. Most of these modules rely on the extraction of intensity gradient-based edges. The following are examples of image-based granulometry systems.

FragScan system. The FRAGSCAN system [33] measures the size distribution of blasted rock, from a dumper or on a conveyor belt, with the help of a camera and a mathematical morphology technique. This system is fully automatic and provides reliable, as well as consistent results, as proven by extensive experimentation. This system is appropriate for industrial use.

WipFrag system. The WipFrag image analysis software [34] analyzes the digital image of the blasted rock with a granulometry system to predict the grain size distribution in the muck pile. Typically, camcorder images of the muck pile are acquired in the field. A scale device is used in each view to reference the sizing. The muck pile is photographed or videotaped, and this image is transferred to the WipFrag system. The broken rock image is transformed into a particle map or network. Network areas are converted into volumes and weights, and the resulting data is displayed as a graph. The fidelity and speed of fragment edge detection allows fully automatic remote monitoring at a rate of one image per three to 5 s. More fragments are resolved, over a greater size range. WipFrag allows comparing the automatically generated net against the rock image. The fragment boundaries are analyzed efficiently using edge detection variables (EDV).

Fragalyst system. Fragalyst is an image analysis system [35] developed by CMRIRegional Centre, Nagpur (India) and the Wavelet Group of Pune (India). This system consists in capturing video photographs of the muck pile and downloading them to the computer or capturing the photos of the muck pile in the field with a digital camera/ordinary camera and then converting the images to grayscale and performing image enhancement, calibration and blob (grain) analysis. This system determines the area, size and shape of the fragments in muck pile/grain aggregates on the basis of grayscale differences.

The 2D information available from the software can further be processed for stereological analysis for 3D information.

3.3. Building Modeling and Segmentation

Within the field of built heritage documentation, automatic delineation is being researched using either optical images [36] or 3D point clouds obtained with terrestrial laser scanners [37,38]. Bienert [39] proposes a method for automatic vectorization based on extracting the contours of the profiles of a building. Briese and Pfeifer [37] have developed algorithms for the automatic extraction of lines that delimit elements in a 3D point cloud, but it still requires manual input. Boulaassal *et al.* [38] combine 3D point clouds obtained with different laser scanners (terrestrial, airborne and vehicle-based) and transform this into a model composed of geometric shapes that represents the building. Nevertheless, none of these methods obtain the delineation, or tracing, we are looking for. Sithole [40], on the other hand, has developed a method to detect bricks in a wall through segmentation of the 3D point cloud. However, his method requires that the mortar joining the bricks is at a significantly different depth from that of the bricks (*i.e.*, the bricks's surfaces and mortar must be in different planes) and that there is sufficient separation between bricks. These assumptions do not hold in many of the stone masonry cases we are analyzing, and thus, this method is irrelevant to us. Demarsin [41] developed a method to extract straight lines delimiting different elements of industrial parts for his thesis. However, he points out that the application of this method to a 3D point cloud of an architectural element would not be appropriate, due to the differences between the point clouds (like the size and the presence of merely ornamental elements). Hammoudi [42] extracts polygon models for façades from 3D point clouds and cadastral maps.

In the context of building façade segmentation using images, Burochin [43] introduces a new unsupervised segmentation method for the extraction of façade main structures, which are characterized by a horizontal and vertical gradient accumulation, which enhances the detection of repetitive structures. Hernandez [44] describes an automatic method for the segmentation of building façade images, where two types of divisions are addressed: intra-façades and inter-façades. The approach introduces several morphological filters to augment the robustness to problems, such as textured balconies and some specular reflection.

4. Proposed Delineation Framework

Our ultimate objective is to delineate stone masonry given its image. If we were able to delineate the individual stones in the whole image, correctly and noise-free, the categorization task for the geometric features in the protocols would be trivial by performing some basic shape analysis on the obtained individual shapes. However, the automatic delineation of individual stones is challenging, if not unfeasible, for built heritage (as discussed in Section 2.2). Therefore, we seek a partial delineation that provides as much information as possible about the geometric configuration of the stones.

This paper proposes a delineation framework that has been developed by the authors using the open source computer vision library, OpenCV [45]. The delineation will be achieved by extracting a set of straight line segments without relying on traditional intensity gradient edge extracting methods.

The first step of this framework is to convert the color image into a grayscale image. Then, the grayscale image is partitioned into regions of interest (ROI) of an (empirically) predetermined size (300 × 300 pixels), starting from the top-left corner of the image, in such way that the ROIs cover the image completely. All ROIs of the partition are processed, each independently, in order to remove outliers and equalize their histogram (Section 4.1) and extract delineating straight segments (Section 4.2). Finally, the results (straight segments) obtained for the ROIs are combined to generate a delineation for the complete image (Section 4.3). Figure 6 provides a schematic description of the proposed delineation framework. Figure 7a shows the original image and an example ROI (marked in red). Figure 7b shows the obtained delineation.

Figure 6. Schematic depiction of the delineation framework. ROI, region of interest.
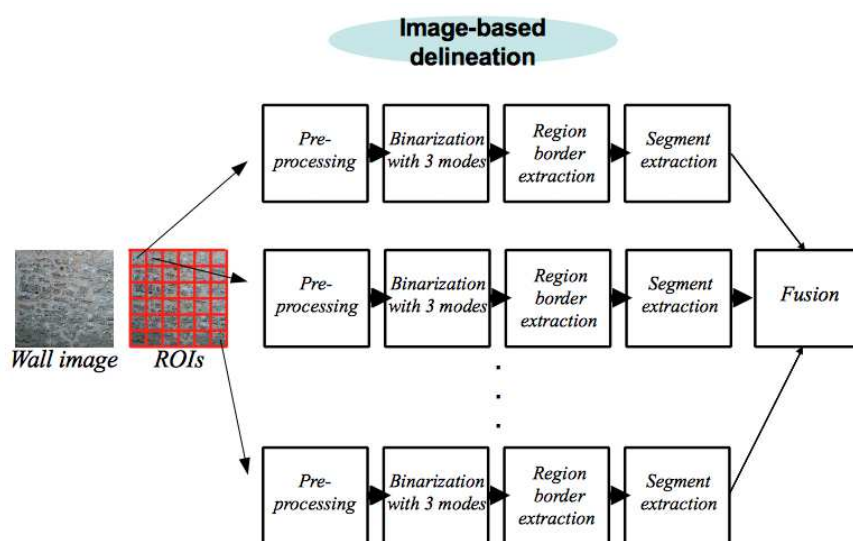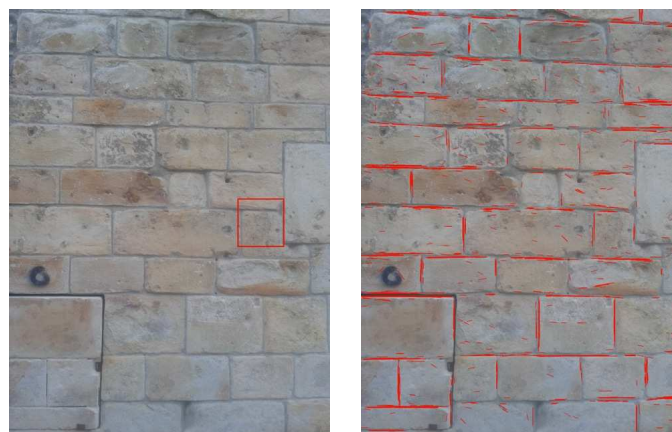

Figure 7. (**a**) Original image (scaled with example regions of interest (ROI) marked in red); (**b**) the obtained delineation (a set of 2D segments marked in red) using the proposed framework.

(**a**) Original　　　　　(**b**) Delineation

4.1. Preprocessing of an ROI

Each grayscale ROI is prepared for processing by removing outliers and equalizing its histogram. Outliers are those intensities with a frequency below some threshold in the intensity histogram. The threshold for removing outliers has been determined empirically and is currently set to 5% of the maximum frequency in the histogram. Outlier pixels are smoothed out by inpainting (a method for removing damaged areas by taking the color and texture at the border of the damaged area and propagating and mixing it inside the damaged area [46]). Then, the histogram is equalized. The images are not processed as a whole, because equalizing the histograms of the ROIs independently helps avoid (to a significant extent) the effects of shade, shadows, bright areas, reflections and similar, for our purposes. Figure 8 shows the main steps related to the processing of a given ROI (in the remainder, to illustrate the preprocessing and processing of an ROI, we will always use the example ROI from Figure 7a). See Figure 8a for our example grayscale ROI and Figure 8b after removing outliers and equalizing its histogram.

4.2. Processing of an ROI

The objective is to use straight segments to delineate the masonry. We have chosen to extract straight segments, because this allows us to obtain additional information, such as slope (the steepness of a straight line), length and position of the segment. The statistics of these geometric properties will be very valuable for subsequent classification tasks. In our framework, the straight 2D segments are not derived from the image gradients, but from the boundaries of some detected regions in the preprocessed images, as explained next.

4.2.1. Region Segmentation Using Most Frequent Intensities

For each ROI, a number of its most frequent intensities are chosen. For each chosen intensity (mode), a binary image is created using an interval of intensities centered on the chosen intensity (mode) and with a certain radius. This binary image can be thought of as a bipartition of the original image. This bipartition will result in a region segmentation. The intuition behind using a bipartition based on several modes is that we can usually find mortar or a dark shadow in between construction blocks, and thus, this image would go a long way toward delineating the blocks. In the case of a delineating shadow, this low intensity will quite often not be one of the most frequent intensities, and thus, it is included by default in the set of modes. In the case of binding mortar being present, the mortar might have been, originally, of a uniform color. However, due to weathering and lighting effects, it will most likely take different intensities throughout the ROI. Additionally, the mortar within different parts of an ROI might belong to different construction phases dating to different periods or different restoration interventions and, thus, have different colors. For these reasons, we use several intensity modes, each resulting in a bipartition, *i.e.*, a region segmentation. In our work, we use three intensity modes (one always being zero, the two remaining modes are chosen from the histogram peaks) and a radius of 50, which was empirically selected. We stress the fact that the number of modes can be more than three. In our case, we find that three modes are enough in order to give accurate results.

By using the three modes and their support in the gray-level scale (which may overlap), one can get three binary images. Figure 8c shows the three binary images for the chosen three modes. These images correspond to three region segmentations. Each of these three binary images is processed independently, and the 2D line segments detected are all saved as part of the delineation set.

Figure 8. The main stages of processing the example ROI in Figure 7a. (**a**) Grayscale; (**b**) preprocessed; (**c**) region segmentation using three modes; (**d**) corresponding boundaries; (**e**) combined, extracted straight segments. (**a**) ROI; (**b**) pre-processed ROI; (**c**) segmentation for intensity of zero; (**c1**) first mode segmentation; (**c2**) second mode segmentation; (**d**) first boundaries map; (**d1**) second boundaries map; (**d2**) third boundaries map; (**e**) extracted 2D segments associated with the three maps in (d).

(**a**)   (**b**)

(**c**)   (**c1**)   (**c2**)

(**d**)   (**d1**)   (**d2**)

(**e**)
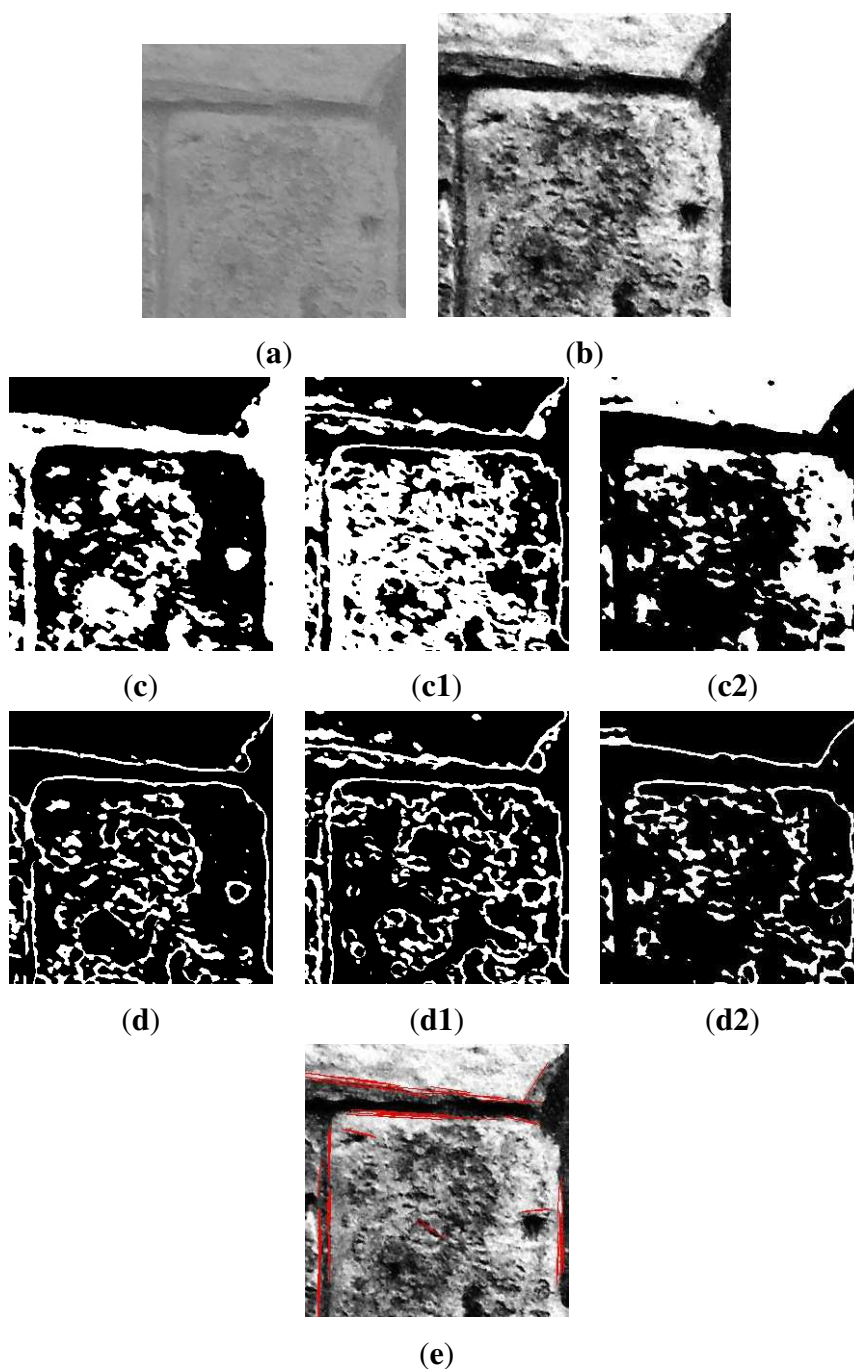
4.2.2. Extracting Boundaries by Removing Inner Patches

Once the image regions are segmented, it will be useful to extract their boundaries. To this end, we remove the inside of solid patches that may be present in the binary images, so that the probabilistic Hough transform (PHT) [47] can segment the corresponding boundaries only. Inner patches are detected with an averaging filter. When this filter is run on a binary image, the result for each pixel is directly proportional to the number of white neighboring pixels (belonging to the segmented region). Those pixels with a result above a certain threshold (depending on the size of the neighborhood used) will be identified as being part of a patch, and they will be removed. The usual method for extracting boundaries is to subtract the eroded binary image from the original. However, our experiments have shown that the method presented in this paper obtains better results for the purpose at hand. Figure 8d shows the extracted boundaries obtained from the three binary images associated with the ROI.

4.2.3. Straight Segment Extraction

Once we get the binary boundaries, we apply a closing operation on them. The probabilistic Hough transform (PHT) (as an implementation of this algorithm, we use the CV_HOUGH_PROBABILISTIC method of the cvHoughLines2 function in OpenCV 2.3.1 [45]) is then used in order to extract the 2D straight line segments. The result of this process for the example ROI can be seen in Figure 8e. In this figure, we can observe that the line segments are close to each other in some parts of the delineation. This redundancy is due to the use of three segmented images that may have region boundaries close to each other. We stress the fact that this redundancy can be beneficial for describing the local geometry of the masonry and, thus, offering discriminative information that will be explained in the next section.

4.3. Fusion of ROI Delineations and Post-Processing

The final delineation is the set of all the segments detected in all three binary images and in all ROIs. The post-processing of the delineation consists in joining collinear segments that are close to each other and removing small segments. Collinear segments are those for which the difference between their polar coordinates is below some predefined threshold. Two segments that have been classified as collinear are joined together if the distance between their closest end points is below some predefined threshold. Joining two segments together means that those two segments are replaced by a new straight segment delimited by the two furthest end points. Lastly, segments whose length is below the mean length have been removed from the set, because their slope is not very reliable, and they are often noisy. We point out that this post-processing step is very useful for overcoming the fragmentation of the detected segments due to the ROI partition of the full image.

5. Results and Performance Evaluation

In this section, we present, first, delineation results for a selected subset of walls and, then, the masonry classification results for the whole data sample based on the obtained delineations.

5.1. Automatic Delineation Results

Figure 7b illustrates the final delineation of the masonry (a set of 2D segments shown in red) associated with the image. Figure 9 illustrates the obtained delineation associated with four, very different walls. The two walls on the right show the regular arrangement of regular blocks, whereas the two on the left show irregular blocks, arranged in rows in the top wall and semi-irregularly in the bottom wall. The top-left wall shows a case in which some sort of mortar has been applied on the center-right area of the wall, partly occluding the boundaries between the stones. The bottom-left wall shows a case in which vegetation is partly occluding the wall.

Figure 9. Examples of delineation results. In each example, the upper part illustrates the original image and the lower part the obtained delineation. We stress the fact that the white regions are formed by a set of 2D segments that can be dense at some locations.

If the task at hand were to classify walls of modern, instead of heritage, buildings, then the delineation task would be simpler. Figure 10 shows the result of applying the proposed delineation method to a modern brick wall (this wall belongs to a modern building, not to built heritage, and is not in our data sample). We can see that there is much less noise in this delineation, and almost all bricks are delineated

correctly. Using the regularity of both the bricks and their arrangement, it would be easy to modify the delineation method to infer the delineation of the missing bricks.

Finally, Figure 11 illustrates a comparison between the Canny filter-based delineation and our proposed delineation. To this end, we chose the best Canny filter (Figure 3a) and then applied PHT with eight different configurations. The best delineations are shown in Figure 11a and Figure 11b. The delineation obtained with the proposed method is illustrated in Figure 11c. As can be seen, while the approach based on the Canny filter was able to provide delineation for some of the stones, many of the stones do not have 2D segments associated with them. On the other hand, our proposed method was able to provide good 2D segments for many stones.

Figure 10. The result of applying the proposed delineation method to a modern brick wall.
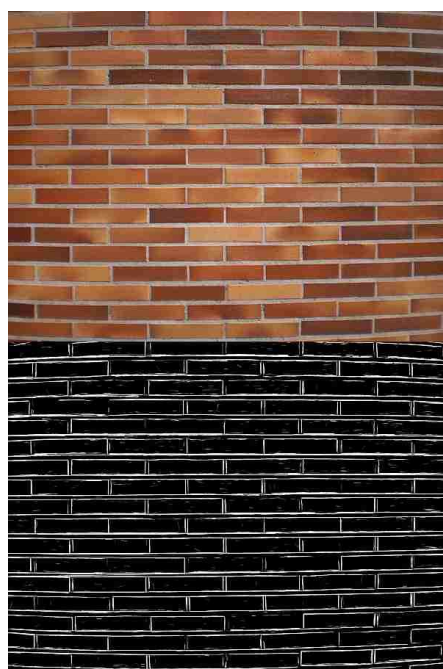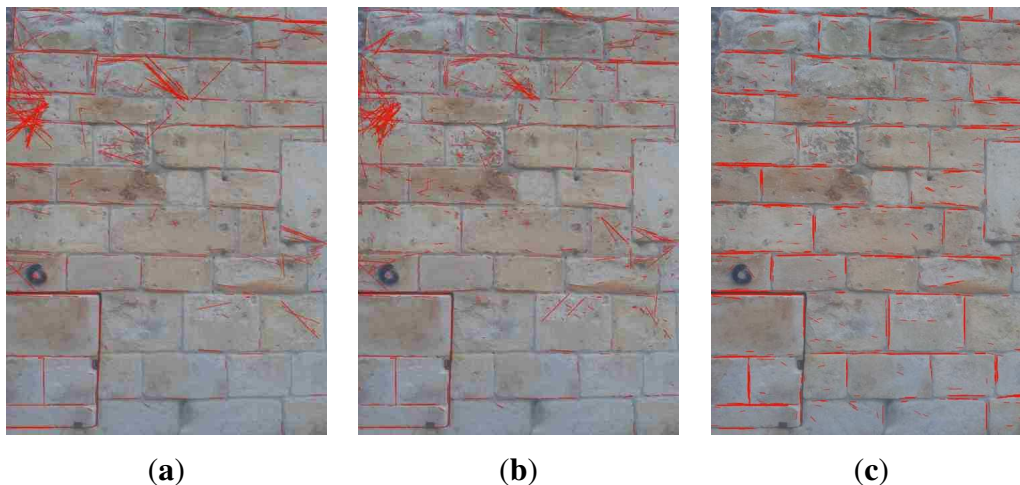

Figure 11. Examples of delineation results. The **(a)** and **(b)** delineations were obtained with the best Canny filter and the best probabilistic Hough transform (PHT) configurations. The **(c)** delineation was obtained with the method proposed in this paper.

(a)   (b)   (c)

5.2. Performance Evaluation: Masonry Classification

This section is structured as follows. Firstly, we give an overview of the classifiers used. Secondly, we give a brief description of feature subset selection. Thirdly, we present the experimental results obtained with the images of masonry walls. Last, we provide a critical analysis of the results. The feature selection and classification results have been obtained using the Waikato Environment for Knowledge Analysis (WEKA) as a machine learning tool [48].

5.2.1. Machine Learning Approaches

Classification is the sub-field of supervised learning that is concerned with the prediction of the category of a given input. The classification model or classifier is trained using a labeled training set (*i.e.*, a dataset containing observations whose category membership is known). Each observation in the dataset is an n-dimensional vector, and each element of the vector is called a *feature* (also *attribute* or *variable*). We have used five classifiers: K nearest neighbor (K-NN) with (K = 1 and K = 3), support vector machines (SVMs), naive Bayes (NB) and classification trees (C4.5). A brief description of all of them is included below.

Instance-Based Learning   Instance-based learning belongs to the K-NN paradigm, a distance-based classifier. It computes the distance of a new case to be classified to each of the observations in the database it uses as the model and decides the class it will assign based on the K nearest cases. We have used the instance-based algorithm described in [49,50].

Support vector machines (SVMs)   SVMs are a set of related supervised learning methods used for classification and regression. In a bi-class problem, SVM views the input data as two sets of vectors (one set per class) in an n-dimensional space. The SVM will construct a separating hyperplane in that space, one which maximizes the margin between the two datasets. To calculate the margin, two parallel hyperplanes are constructed, one on each side of the separating hyperplane, which are "pushed up against" the two datasets. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the neighboring data points of both classes, since, in general, the larger the margin, the lower the generalization error of the classifier [51]. SVMs were extended to classify datasets that are not linearly separable through the use of non-linear kernels. In our work, we use non-linear SVMs with a radial kernel.

Probabilistic classifiers   There are many classifiers based on probability theory. Most of them are based on Bayes' theorem and try to obtain the class for which the *a posteriori* probability is the greatest given the predictor variables of the case to be classified. In this work, we have used the naive Bayes (NB) classifier [52]. The name of this classifier comes from its underlying assumption, namely that the features are independent. This is a very strong assumption and, hence, considered "naive".

Classification trees   A classification tree is a classifier composed of nodes and branches that break the set of samples into a set of covering decision rules. In each node, a single test is made to obtain the

partition. The starting node is called the root of the tree. In the final nodes or leaves, a decision about the classification of the case is made. In this work, we have used the C4.5 algorithm [53].

5.2.2. Feature Selection

Feature selection (FS) is the process of identifying the features that are relevant to a particular learning (or data mining) problem. FS is a key process in supervised classification [54] and can improve classification performance (accuracy, area under the receiver operating characteristic (ROC) curve (AUC), *etc.*).

Most algorithms for FS can be categorized as filter or wrapper approaches. In the filter approach, an attribute (or attribute subset) is evaluated by only using intrinsic properties of the data (e.g., statistical or information-based measures). Filter techniques have the advantage of being fast and general, in the sense that the subset obtained is not biased in favor of a specific classifier. However, they lack robustness against interactions among features. In addition, it is not clear how to determine the cut-off point for rankings to select only truly important features and exclude noise.

Wrapper algorithms have the advantage of achieving greater accuracy than filters, but with the disadvantage of being (far) more time-consuming and obtaining an attribute subset that is biased towards the used classifier. Over the last decade, wrapper-based FS has been an active area of research. Different search algorithms [55] have been used to guide the search process, while some classifiers (e.g., naive Bayes, KNN, *etc.*) are used as a surrogate in order to evaluate the goodness of the subset proposed by the search algorithm.

5.2.3. Experimental Results

Data Preparation   As presented in Section 4, each raw image will undergo the proposed delineation steps that provide a set of 2D straight segments. Several statistics are extracted from this set of segments and are used as the predictor variables in the classifier. The statistics extracted are listed next, where the lengths are expressed as a percentage of the image width, the slopes represented in degrees and the slope differences the differences between slopes for every pair of segments in the delineation. For the image of the processed delineation, the sums of the rows (horizontal accumulation) and the sums of the columns (vertical accumulation) are calculated.

- Regarding the lengths, these statistics are the minimum, maximum, least frequent, most frequent, mean and standard deviation.
- Regarding the slopes, the statistics used are the maximum, least frequent, most frequent, mean, standard deviation and percentage of slopes that are vertical (or very nearly vertical, *i.e.*, a 90-degree slope).
- For slope differences, the statistics used are the least frequent, most frequent, mean, standard deviation, percentage of differences between zero and four degrees (pairs of segments that are parallel or nearly parallel) and percentage of differences between 86 and 94 degrees (pairs of segments that are perpendicular or nearly perpendicular).
- For the horizontal accumulation, the maximum (expressed as the percentage of the width), least frequent, mean and standard deviation are used.

- For the vertical accumulation, the minimum (expressed as the percentage of the height), maximum, least frequent, most frequent, mean and standard deviation are used.

Altogether, every image is described by 28 features that summarize statistics about the extracted straight segments. Every feature was rescaled to the interval [0, 1].

Classifier Parameters The classifiers we have chosen to use are the K Nearest Neighbor (K-NN), Support Vector Machines (SVM), Naive Bayes (NB), and Classification Trees (C4.5) algorithms. For K-NN we have chosen to test two possible values of K, the number of nearest neighbors, 1 and 3. For SVM, we use a non-linear SVM with a Gaussian kernel $e^{\left(-\gamma |u-v|^2\right)}$ with $\gamma = 1/N = 1/86$ (N being the number of instances) and the cost parameter C for C-SVC equal to 150.

Classification Using All Features and Manually Selected Features The performance of a classifier is usually measured in terms of the success rate (or classification accuracy); that is, the proportion of instances correctly classified given a test set. The test set must contain instances that are not present in the training set, *i.e.*, the dataset used to train the classifier. When the available dataset is small, splitting it into training and test sets means the classifier has to be trained and tested, respectively, with few data. In order to attenuate this problem, we have chosen to follow the leave-one-out cross-validation (LOOCV) protocol to perform the evaluation. The k-fold cross-validation technique, in general, splits the dataset into k subsets. Then, it performs k iterations of training and testing, using one of the k subsets as the test set (a different one each iteration) and the union of the remaining subsets as the training set. After the k iterations, each instance of the dataset has been classified by the classifier exactly once. These predicted classifications are then compared to the actual class labels to calculate the success rate. The LOOCV technique is a particular case of the k-fold cross-validation technique in which k is set to the number of instances in the data. This, of course, means that each fold contains just one element.

In order to quantify the classification accuracy, we have performed LOOCV tests using the five classifiers. Table 1 illustrates the classification accuracy obtained with the five classifiers in two cases. The first case corresponds to the use of the set of the 28 extracted features. The second case corresponds to the use of a subset of 19 manually selected features out of the 28. These 19 features are of two types: original features and features that were formed by combining original features. As can be seen, by using the raw 28 features, the non-linear SVM provided the best performance. However, when the manually selected features were used, the 1-NN provided the best performance. We can also observe that the use of manually selected features improved the performance of all classifiers (except for the BN classifier for which the performance remains the same). For example, the SVM performance increased 2.7%.

Classification with Automatic Feature Selection The above manual selection of features was guided by some intuition about feature relevance in discriminating classes. However, this manual selection is not necessarily optimal and can be replaced by automatic feature selection using, for example, the wrapper technique. The goal is to find a subset among the complete set of features (28 features) that maximizes the classification accuracy. We evaluated the performance using LOOCV. The search strategy chosen was a genetic algorithm [56] with the following parameter values: the population size was set to 80; the maximum number of generations was set to 20; the crossover probability was set to 0.7; and the mutation

probability to 0.05. This kind of processing is carried out efficiently, since the total number of features is 28.

Table 1. Overall classification results using all the extracted features (28 features) and the manually selected subset (19 features). NN, nearest neighbor; NB, naive Bayes; SVM, support vector machine.

		28 Features		*19 Features*	
Classifier	**Cases**	**No. of Errors**	**Accuracy**	**No. of Errors**	**Accuracy**
1-NN	86	24	72.1%	17	80.2%
3-NN	86	22	74.4%	21	75.6%
NB	86	25	70.9%	25	70.9%
J48	86	26	69.8%	23	73.3%
SVM	86	21	75.6%	19	78.3%

Table 2 shows the classification accuracy obtained with the five classifiers and the features selected by the wrapper method. Compared to the scheme that uses all features (28 features) or the manually selected features (19 features), the accuracy of all classifiers was improved. For example, the recognition rate of the 1-NN classifier increased by 12.8% with respect to the result with 28 features and by 4.2% with respect to the result with the manually selected features (19 features). We can observe that, following automatic feature selection, the SVM classifier provided the best performance.

Table 2. Classification results using automatic feature selection and leave-one-out cross-validation (LOOCV) evaluation.

Classifier	**Cases**	**No. of Errors**	**Accuracy**
1-NN	86	13	84.9%
3-NN	86	14	83.7%
NB	86	14	83.7%
J48	86	12	86.0%
SVM	86	11	87.2%

Table 3 illustrates the precision, recall and F1 measure for every class obtained after feature selection for all classifiers. Finally, Table 4 shows the confusion matrix for the best result obtained, that is, the SVM classifier with feature selection using the wrapper method.

Table 3. Recall, precision and F1 measure for all classes and for all classifiers.

Classifier	*Class 1*			*Class 2*			*Class 3*		
	Rec.	**Prec.**	**F1**	**Rec.**	**Prec.**	**F1**	**Rec.**	**Prec.**	**F1**
1-NN	84.8%	87.5%	86.2%	80.0%	66.7%	72.7%	86.6%	91.7%	89.2%
3-NN	97.0%	78.0%	86.5%	53.3%	80.0%	64.0%	84.2%	91.4%	87.7%
NB	97.0%	84.2%	90.1%	60.0%	64.3%	62.1%	81.6%	91.2%	86.1%
J48	90.9%	96.8%	93.7%	73.3%	68.8%	71.0%	86.8%	84.6%	85.7%
SVM	97.0%	82.1%	88.9%	73.3%	78.6%	75.9%	84.2%	97.0%	90.1%

Table 4. Confusion matrix from the SVM classifier with automatic feature selection.

	Classified as		
	Class 1	**Class 2**	**Class 3**
Class 1	32	1	0
Class 2	3	11	1
Class 3	4	2	32

Note that the wrapper technique for feature selection makes use of the classifier to evaluate the different feature subsets. Thus, the optimal subset of features differs from one classifier to another. Table 5 illustrates the selected features for every classifier. For every feature, we show a binary string formed by five bits that correspond to the five classifiers, 1-NN, 3-NN, NB, J48 and SVM, respectively. A zero bit indicates that the feature was not selected by the wrapper technique for the corresponding classifier. A one bit indicates that the feature was selected. As expected, the relevance of features depends highly on the classifier used. For instance, the SVM classifier (fifth bit) preferred one feature related to lengths, four features related to slopes and their differences, two features related to the percentage of pairwise relation (parallel and perpendicular) and five features about row and column accumulation. Moreover, we can observe that all classifiers have selected the amount of segment pairs having a parallelism relation.

Analysis of the Results   The results show that all classifiers have more difficulty classifying the instances of the second class correctly. The reason is twofold. First, this is the class with the least number of training examples, and this has a direct negative impact on the performance of the classifier. Second, the geometric arrangement of walls belonging to Class 2 shares many statistics with the two remaining classes. This is because Class 2 is, by definition, the class in between Classes 1 and 3. Class 2 shares the use of non-regular blocks with Class 1. On the other hand, Class 2 shares with Class 3 that the blocks are arranged in rows.

In practice, discriminating between Classes 1 and 2 is challenging, even for humans; the boundary is not at all clear-cut. Figure 12 shows a Class 2 wall that could be easily classified as Class 1. The blocks in this wall are irregular, so it is clearly not Class 3. However, we could say that the bottom half, in

which we can clearly see rows, is Class 2 and the top half, in which there are no obvious rows, Class 1. Therefore, misclassifying Class 1 and 2 instances is not, generally, a fatal error. Additionally, in some cases, even misclassifying Class 2 and 3 instances might not be fatal.

However, misclassifying Class 1 and 3 instances is always a fatal error. From the confusion matrix in Table 4, we can conclude that the best performing classifier has made only four fatal errors (4.7%), and these are all Class 3 instances that have been classified as Class 1. The fatal errors are due to the incorrect delineation of these images. Figure 13 shows an example in which the delineation framework proposed almost completely fails to delineate the regular blocks in a Class 3 wall. This is, currently, the weakest point of the framework and one to which it is imperative that we find a solution.

Table 5. Features selected by the wrapper method.

Feature	Selected/Not Selected
1 minimum length	00010
2 maximum length	10100
3 least frequent length	00010
4 most frequent length	00110
5 length mean	11010
6 length standard deviation	01011
7 maximum slope	00110
8 least frequent slope	10101
9 most frequent slope	01010
10 slope mean	00011
11 slope standard deviation	01100
12 percentage of vertical slopes	00100
13 least frequent slope difference	00001
14 most frequent slope difference	00010
15 slope difference mean	00011
16 slope difference standard deviation	00110
17 percentage of pairs of segments nearly parallel	11111
18 percentage of pairs of segments nearly perpendicular	11101
19 maximum row count	00101
20 least frequent row count	10010
21 row count mean	10001
22 row count SD	01000
23 minimum column count	01011
24 maximum column count	10010
25 least frequent column count	10101
26 most frequent column count	11101
27 column count mean	10000
28 column count SD	01000

Figure 12. Class 2 wall.

Figure 13. Example of Class 3 misdelineation.
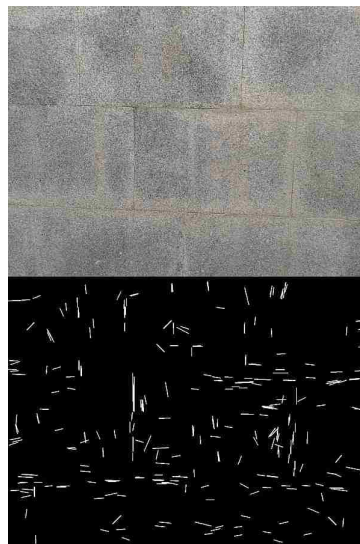
In conclusion, the results obtained are promising. The best classifier achieved a 87.2% success rate, and among the 12.8% of incorrectly classified cases, only four (4.7% of the sample size) were fatal errors. Nonetheless, the delineation algorithms need to be improved so that, at least, the classifier makes no fatal errors.

6. Conclusions and Future Work

This paper has presented the foundational work carried out for the development of a built heritage analysis and classification ICT tool. The goal is to automate the application of new protection protocols under development with the objective of speeding up the protection process. The new framework introduced in this paper for the automatic delineation of the stone masonry has been evaluated using a classification task. The results obtained are very satisfactory and show that the obtained delineation provides meaningful information.

Since individual stone delineation is infeasible, we have proposed a heuristic that uses image region boundaries using image quantization (similar to the way Fisher clustering is based on histograms). These boundaries are then transformed to a set of straight segments that describe the geometric arrangement of blocks in a given wall. Based on statistics related to these segments, walls can be automatically classified

using state-of-the-art classifiers. We have incorporated a wrapper technique in order to select the most relevant features (statistics). Five different classifiers were tested. The results obtained are promising.

The paper scope was extracting useful information from images of masonry. While any image processing tool can be affected by the stone lithology variability, our main goal was to provide a generic purpose ICT tool that can be useful for a wide range of the lithology variability. Indeed, the studied cases have a lot of variability regarding textures, sizes, surface state, mortars, *etc*.

Future work will focus on improving the delineation by removing as many noisy segments as possible; noisy, in this context, meaning all segments that do not belong to the delineation. The use of RGB-D cameras could be very useful. This has the advantage that lighting conditions, shadows and reflections have no effect on the depth map. The joint use of image intensity and depth maps can increase the ability to detect more useful segments that provide better delineation. Furthermore, future work may investigate the use of probabilistic color and texture gradient maps (similarly to [57]) that can be used as global descriptors. The other research line will focus on building a subspace by mapping the statistics. The purpose of using these subspaces is to enhance class discrimination.

Acknowledgments

Author Contributions

Noelia Oses acquired the image data, developed the delineation methods, and implemented all image delineation processes. Fadi Dornaika developed the main idea that led to this paper and wrote the paper together with Noelia Oses. Abdelmalik Moujahid addressed the evaluation part and performed all the feature selection and classification tasks.

Conflicts of Interest

The authors declare no conflicts of interest.

References

1. Kersten, T.P.; Lindstaedt, M. Image-based low-cost systems for automatic 3D recording and modeling of archaeological finds and objects. *Lect. Note. Comput. Sci.* **2012**, *7616*, 1–10.
2. Moussa, W.; Abdel-Wahab, M.; Fritsch, D. Automatic fusion of digital images and laser scanner data for heritage preservation. *Lect. Note. Comput. Sci.* **2012**, *7616*, 76–85.
3. Stanco, F. *Digital Imaging for Cultural Heritage Preservation: Analysis, Restoration, and Reconstruction of Ancient Artworks*; CRC Press: Boca Raton, FL, USA, 2011.
4. Izkara, J.; Mediavilla, A.; Rodriguez-Maribona, I. Information Technologies and Cultural Heritage: Innovative Tools for The Support in the Participative Management of Historical Centres. In Proceedings of the Conference on Sustaining Europe's Cultural Heritage, Ljubljana, Slovenia, 10–12 November 2008.

5. Calabrò, E. Applying photogrammetric techniques to determine facade decays: The case study of zisa palace, Italy. *Engineering* **2012**, *4*, 707–712.

6. Zaccarini, M.; Iannucci, A.; Orlandi, M.; Vandini, M.; Zambruno, S. A Multi-Disciplinary Approach to the Preservation of Cultural Heritage: A Case Study on the Piazzetta Degli Ariani, Ravenna. In Proceedings of IEEE Digital Heritage, Marseille, France, 28 October–1 November 2013.

7. Alshawabkeh, Y.; El-Khalili, M. Detection and quantification of material displacements at historical structures using photogrammetry and laser scanning techniques. *Mediterr. Archaeol. Archaeom.* **2013**, *13*, 57–67.

8. Stefani, C.; Brunetaud, X.; Janvier-Badosa, S.; Beck, K.; Luca, L.D.; Al-Mukhtar, M. 3D information system for the digital documentation and the monitoring of stone alteration. *Lect. Note. Comput. Sci.* **2012**, *7616*, 330–339.

9. Godin, G.; Beraldin, J.A.; Taylor, J.; Cournoyer, L.; Rioux, M.; El-Hackim, S.; Baribeau, R.; Blais, F.; Boulanger, P.; Domey, J.; *et al.* Active optical 3-D imaging for heritage applications. *IEEE Comput. Graph. Appl.* **2002**, *22*, 24–36.

10. Muller, P.; Wonka, P.; Haegler, S.; Ulmer, A.; Gool, L.V. Procedural modeling of buildings. *ACM Trans. Graph.* **2006**, *25*, 614–623.

11. Pasko, G.; Pasko, A.; Vilbrandt, T.; Filho, A.; da Silva, J. Digital interpretation of cultural heritage: 3D modeling and materialization of 2D artworks for future museums. *Int. J. Incl. Mus.* *3*, 63–80.

12. Remondino, F. Heritage recording and 3D modeling with photogrammetry and 3D scanning. *Remote Sens.* **2011**, *3*, 1104–1138.

13. Levoy, M.; Pulli, K.; Curless, B.; Rusinkiewicz, S.; Koller, D.; Pereira, L.; Ginzton, M.; Anderson, S.; Davis, J.; Ginsberg, J.; *et al.* The Digital Michelangelo Project: 3D Scanning of Large Statues. In Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '00, New York, USA, 23–28 July 2000; pp. 131–144.

14. Ikeuchi, K.; Oishi, T.; Takamatsu, J.; Sagawa, R.; Nakazawa, A.; Kurazume, R.; Nishino, K.; Kamakura, M.; Okamoto, Y. The great buddha project: Digitally archiving, restoring, and analyzing cultural heritage objects. *Int. J. Comput. Vision* **2007**, *75*, 189–208.

15. Riveiro, B.; Morer, P.; Arias, P.; de Arteaga, I. Terrestrial laser scanning and limit analysis of masonry arch bridges. *Constr. Build. Mater.* **2011**, *25*, 1726–1735.

16. Laefer, D.F.; Truong-Hong, L.; Carr, H.; Singh, M. Crack detection limits in unit based masonry with terrestrial laser scanning. *NDT & E Int.* **2014**, *62*, 66–76.

17. Tapete, D.; Casagli, N.; Luzi, G.; Fanti, R.; Gigli, G.; Leva, D. Integrating radar and laser-based remote sensing techniques for monitoring structural deformation of archaeological monuments. *J. Archaeol. Sci.* **2013**, *40*, 176–189.

18. Rodríguez, A.; Valle, J.; Martínez, J. 3D Line Drawing from Point Clouds Using Chromatic Stereo and Shading. In Proceedings of the 14th International Conference on Virtual Systems and Multimedia, Limassol, Cyprus, 20–25 October 2008; pp. 77–84.

19. Arias, P.; Ordóñez, C.; Lorenzo, H.; Herraez, J. Methods for documenting historical agro-industrial buildings: A comparative study and a simple photogrammetric method. *J. Cult. Herit.* **2006**, *7*, 350–354.

20. Fuentes, J.M. Methodological bases for documenting and reusing vernacular farm architecture. *J. Cult. Herit.* **2010**, *11*, 119–129.

21. Alba, M.I.; Barazzetti, L.; Scaioni, M.; Rosina, E.; Previtali, M. Mapping infrared data on terrestrial laser scanning 3D models of buildings. *Remote Sens.* **2011**, *3*, 1847–1870.

22. De Luca, L.; Busayarat, C.; Stefani, C.; Véron, P.; Florenzano, M. A semantic-based platform for the digital analysis of architectural heritage. *Comput. Graph.* **2011**, *35*, 227–241.

23. Oses, N.; Dornaika, F. Image-based delineation of built heritage masonry for automatic classification. *Lect. Note. Comput. Sci.* **2013**, *7950*, 782–789.

24. Oses, N.; Azkarate, A. New Protocols for Built Heritage Protection in the Basque Country: Towards an Automatic Analysis Tool for Built Heritage. In *Science and Technology for the Conservation of Cultural Heritage*; Lazzari, M., Cano, E., Rogerio-Candelera, M., Eds.; CRC Press: Boca Raton, FL, USA, 2013.

25. ICOMOS-International Scientific Committee for Stone (ISCS). *Illustrated Glossary on Stone Deterioration Patterns*; Monuments and Sites Volume XV; ICOMOS: Paris, France, 2008.

26. Kimmel, R.; Bruckstein, A. On regularized Laplacian zero crossings and other optimal edge integrators. *Int. J. Comput. Vision* **2003**, *53*, 225–243.

27. Zhang, W.; Qin, Z.; Wan, T. Image Scene Categorization Using Multi-Bag-of-Features. In Proceedings of the International Conference on Machine Learning and Cybernetics, Guilin, China, 10–13 July 2011.

28. Duan, K.; Parikh, D.; Crandall, D.; Grauman, K. Discovering Localized Attributes for Fine-Grained Recognition. In Proceedings of the 2012 Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012.

29. Yao, B.; Bradski, G.; Fei-Fei, L. A Codebook-Free and Annotation-Free Approach for Fine-Grained Image Categorization. In Proceedings of the 2012 Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012.

30. Chen, L.; Yan, L. LBP Displacement Local Matching Approach for Human Face Recognition. In Proceedings of the International Workshop on Computer Vision with Local Binary Pattern Variants, ACCV, Daejeon, Korea, 5–6 November 2012.

31. He, S.; Sang, N.; Gao, C. Pyramid-Based Multi-Structure Local Binary Pattern for Texture Classification. In Proceedings of the Asian Conference on Computer Vision, Queenstown, New Zealand, 8–12 November 2010.

32. Wang, W. Rock Particle Image Segmentation and Systems. In *Pattern Recognition Techniques, Technology and Applications*; I-Tech: Vienna, Austria, 2008; pp. 197–226.

33. Schleifer, J.; Tessier, B. FRAGSCAN: A Tool to Measure Fragmentation of Blasted Rock. In *Measurement of Blast Fragmentation*; Balkema: Rotterdam, The Netherlands, 1996; pp. 73–78.

34. Maerz, N.H.; Palangio, T.C.; Franklin, J.A. WipFrag Image Based Granulometry System. In Proceedings of the FRAGBLAST 5 Workshop on Measurement of Blast Fragmentation, Montreal, QC, Canada, 23–24 August 1996.

35. Raina, A.K.; Choudhury, P.B.; Ramulu, M.; Chrakraborty, A.K.; Dudhankar, A.S. Fragalyst—An indigenous digital image analysis system for grain size measurement in mines. *J. Geol. Soc. India* **2002**, *59*, 561–569.

36. Wendel, A.; Donoser, M.; Bischof, H. Unsupervised Facade Segmentation Using Repetitive Patterns. In Proceedings of the 32th Annual Symposium of the German Association for Pattern Recognition (DAGM), Darmstadt, Germany, 21 September 2010.

37. Briese, C.; Pfeifer, N. Towards automatic feature line modeling from terrestrial laser scanner data. *Int. Arch. Photogr. Remote Sens. Spat. Inf. Sci.* **2008**, *XXXVII*, 463–468.

38. Boulaassal, H.; Landes, T.; Grussenmeyer, P. Reconstruction of 3D vector models of buildings by combination of ALS, TLS and VLS data. *Int. Arch. Photogr. Remote Sens. Spat. Inf. Sci.* **2011**, *XXXVIII*, 239–244.

39. Bienert, A. Vectorization, Edge Preserving Smoothing and Dimensioning of Profiles in Laser Scanner Point Clouds. In Proceedings of XXIst ISPRS Congress, Beijing, China, 3–11 July 2008.

40. Sithole, G. Detection of bricks in a masonry wall. *Int. Arch. Photogr. Remote Sens. Spat. Inf. Sci.* **2008**, *XXXVII*, 567–572.

41. Demarsin, K. Extraction of Closed Feature Lines from Point Clouds Based on Graph Theory. Ph.D. Thesis, Katholieke Universiteit Leuven, Leuven, Belgium 2009.

42. Hammoudi, K.; Dornaika, F.; Paparoditis, N. Generating Virtual 3D Model of Urban Street Facades by Fusing Terrestrial Multi-Source Data. In Proceedings of the IEEE International Conference on Intelligent Environments, Nottingham, UK, 25–28 July 2011.

43. Burochin, J.; Tournaire, O.; Paparoditis, N. An Unsupervised Hierarchical Segmentation of a Facade Building Image in Elementary 2D Models. In Proceedings of the ISPRS Workshop on Object Extraction for 3D City Models, Road Databases and Traffic Monitoring, Paris, France, 3-4 September 2009.

44. Hernandez, J.; Marcotegui, B. Morphological Segmentation of Building Facade Images. In Proceedings of the 16th IEEE International Conference on Image Processing, Cairo, Egypt, 7–10 November 2009.

45. Bradski, G. The OpenCV library. *Dr. Dobb's J. Softw. Tools* **2000**, *120*, 122–125. Available online: http://www.drdobbs.com/open-source/the-opencv-library/184404319 (accessed on 18 February 2014).

46. Telea, A. An image inpainting technique based on the fast marching method. *J. Graph. Tools* **2004**, *9*, 25–36.

47. Kiryati, N.; Eldar, Y.; Bruckstein, A. A probabilistic Hough transform. *Pattern Recognit.* **1991**, *24*, 303–316.

48. Mark, H.; Eibe, F.; Geoffrey, H.; Bernhard, P.; Peter, R.; Ian, H.W. The WEKA data mining software: An update. *SIGKDD Explor.* **2009**, *11*, 10–18.

49. Aha, D.; Kibler, D.; Albert, M. Instance-based learning algorithms. *Mach. Learn.* **1991**, *6*, 37–66.

50. Mena-Torres, D.; Aguilar-Ruiz, J.; Rodriguez, Y. An Instance Based Learning Model for Classification in Data Streams with Concept Change. In Proceedings of the 2012 11th Mexican International Conference on Artificial Intelligence (MICAI), San Luis Potosi, Mexico, 27 Octobe–4 November 2012; pp. 58–62.

51. Meyer, D.; Leisch, F.; Hornik, K. The support vector machine under test. *Neurocomputing* **2003**, *55*, 169–186.

52. Zhang, H. The Optimality of Naive Bayes. In Proceedings of FLAIRS Conference, Miami Beach, FL, USA, 17–19 May 2004.

53. Quinlan, J. *C4.5: Programs for Machine Learning*; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2003; Volume 55, pp. 169–186.

54. Liua, H.; Suna, J.; Liua, L.; Zhang, H. Feature selection with dynamic mutual information. *Pattern Recognit.* **2009**, *43*, 1330–1339.

55. ElAlami, M. A filter model for feature subset selection based on genetic algorithm. *Knowl.-Based Syst.* **2009**, *22*, 356–362.

56. Mitchell, M. *An Introduction to Genetics Algorithms*; MIT Press: Cambridge, MA, USA, 1998.

57. Arbelaez, P.; Maire, M.; Fowlkes, C.; Malik, J. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 898–916.