

Article

An Advanced Pre-Processing Pipeline to Improve Automated Photogrammetric Reconstructions of Architectural Scenes

Marco Gaiani ^{1,*}, Fabio Remondino ², Fabrizio I. Apollonio ¹ and Andrea Ballabeni ¹

¹ Department of Architecture, University of Bologna, Bologna 40136, Italy; fabrizio.apollonio@unibo.it (F.I.A.); andrea.ballabeni@unibo.it (A.B.)

² 3D Optical Metrology (3DOM) Unit, Bruno Kessler Foundation (FBK), Trento 38123, Italy; remondino@fbk.eu

* Correspondence: marco.gaiani@unibo.it; Tel./Fax: +39-51-209-3165 (ext. 123)

Academic Editors: Diego Gonzalez-Aguilera, Norman Kerle and Prasad S. Thenkabail

Received: 1 December 2015; Accepted: 25 January 2016; Published: 25 February 2016

Abstract: Automated image-based 3D reconstruction methods are more and more flooding our 3D modeling applications. Fully automated solutions give the impression that from a sample of randomly acquired images we can derive quite impressive visual 3D models. Although the level of automation is reaching very high standards, image quality is a fundamental pre-requisite to produce successful and photo-realistic 3D products, in particular when dealing with large datasets of images. This article presents an efficient pipeline based on color enhancement, image denoising, color-to-gray conversion and image content enrichment. The pipeline stems from an analysis of various state-of-the-art algorithms and aims to adjust the most promising methods, giving solutions to typical failure causes. The assessment evaluation proves how an effective image pre-processing, which considers the entire image dataset, can improve the automated orientation procedure and dense 3D point cloud reconstruction, even in the case of poor texture scenarios.

Keywords: pre-processing; denoise; enhancement; 3D reconstruction; photogrammetry; image matching; automation

1. Introduction

In the last years, the image-based pipeline for 3D reconstruction purposes has received large interest leading to fully automated methodologies able to process large image datasets and deliver 3D products with a level of detail and precision variable according to the applications [1–3] (Figure 1). Certainly, the integration of automated computer vision algorithms with reliable and precise photogrammetric methods is nowadays producing successful (commercial and open) solutions (often called Structure from Motion (SfM)) for automated 3D reconstructions from large image datasets [4–6].



Figure 1. Examples of image-based 3D reconstruction results of complex architectural scenarios achieved with the presented methodology.

For terrestrial applications, the level of automation is reaching very high standards and it is increasing the impression that few randomly acquired images (or even found on the Internet) and a black-box tool (or mobile app) are sufficient to produce a metrically precise 3D point cloud or textured 3D model. Such tools are able to ingest and process large quantities of images almost always delivering an apparently successful solution, which is often a local minimum and not the fully correct one (Figure 2). However, non-expert users might not be able to spot such small errors or divergences in the bundle adjustment due to the fact that only a message of successful image orientation is provided, without statistical analyses.

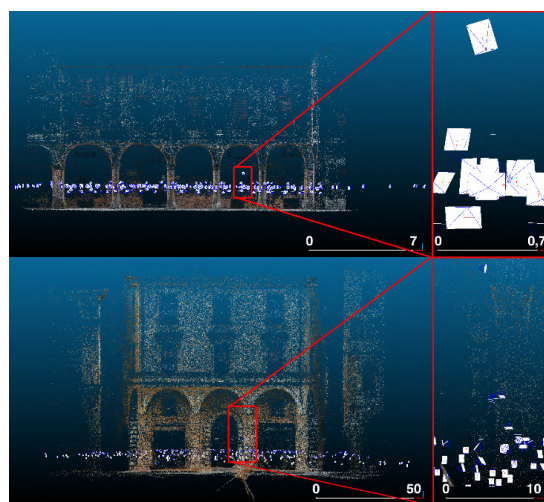


Figure 2. Typical examples of SfM results where, despite a message of successful image orientation and a very small re-projection error, there are some cameras wrongly oriented. A non-expert user could only spot such errors with difficulty and would proceed to the successive processing stages, negatively affecting the final results.

Motion blur, sensor noise and jpeg artifacts are just some of the possible image problems that are negatively affecting automated 3D reconstruction methods. These problems are then coupled with lack of texture scenarios, repeated patterns, illumination changes, *etc.* Therefore, image pre-processing methods are fundamental to improve the image quality for successful photogrammetric processing. Indeed, as the image processing is fully automated, the quality of the input images, in terms of radiometric quality as well as network geometry, is fundamental for a successful 3D reconstruction.

This paper presents an efficient image pre-processing methodology developed to increase the processing performances of the two central steps of the photogrammetric pipeline, *i.e.*, image orientation and dense image matching. The main idea is to minimize typical failure caused by Scale-Invariant Feature Transform (SIFT)-like algorithms [7] due to changes in the illumination conditions or low contrast blobs areas and to improve the performances of dense image matching methods [8]. The methodology tries to: (i) increase the number of correct image correspondences, particularly in textureless areas; (ii) track image features along the largest number of images to increase the reliability of the computed 3D coordinates; (iii) correctly orient the largest number of images; (iv) deliver sub-pixel accuracy at the end of the bundle adjustment procedure; and (v) provide dense, complete and noise-free 3D point clouds (Figure 3). The work investigated various state-of-the-art algorithms aiming to adapt the most promising methods and give solutions at the aforementioned specific problems, thus creating a powerful solution to radiometrically improve the image quality of an image datasets. The developed procedure for image pre-processing and enhancement consists of color balancing (Section 2), image denoising (Section 3), color-to-gray conversion (Section 4) and image content enrichment (Section 5). The pre-processing methodology could be really useful in the architectural, built heritage and archaeological fields, where automated 3D modeling procedures have become very common whereas skills in image acquisition and data processing are often missing. Our methodology could even be embedded in different types of processing scenarios, like completely automated web-based applications (e.g., Autodesk ReCap and Arc 3D Webservice) or offline desktop-based applications (e.g., Agisoft Photoscan, Photometrix iWitness, VisualSFM, and Pix4D Mapper).



(a) Without Pre-Processing

Figure 3. Cont.



(b) With Pre-Processing

Figure 3. 3D reconstruction of complex architectures without (a) and with (b) a suitable pre-processing methodology applied to improve the radiometric image quality and provide more complete and noise-free 3D point clouds.

The pipeline (Figure 4) is afterwards presented and evaluated using some datasets of architectural scenarios.

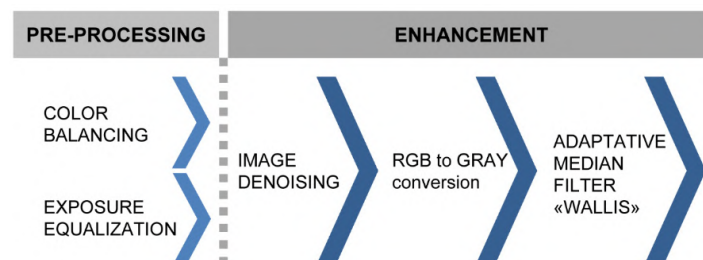


Figure 4. The proposed image pre-processing and enhancement pipeline.

Related Works

Image pre-processing is a set of methods used to increase the quality of images for successive processing purposes [9]. The aim is thus to enhance some image features important, e.g., for 3D reconstruction algorithms or to remove unwanted disturbs or degradations in the image. A pre-processing can be a simple histogram's stretching or a more complex approach like denoising or filtering [10,11]. Image pre-processing normally comprises enhancement (*i.e.*, the improvement of the image quality) and restoration (*i.e.*, the removal of degraded areas). The former is more subjective, whereas the latter is an objective process that involves the modeling of degradation (possibly from prior knowledge) and applying an inverse process to recover the original signal. Image pre-processing is a fundamental task for many successive tasks in applications like medical imaging [12], computer vision, underwater photogrammetry [13] or 3D modeling [14].

Maini and Aggarwal [15] provide an overview of concepts and algorithms commonly used for image enhancement. Stamos *et al.* [16] presents some metrics to estimate the amount of blur in image sequence, based on color saturation, local auto-correlation and gradient distribution. Feature tracking and camera poses recovery methods in blurry image sequences can be improved using edgelets [17] or blurring the previous frame in order to obtain a consistent tracking [18] or deblurring a current frame with a blur kernel [19]. Guidi *et al.*, [14] analyses how image pre-processing with polarizing filters and HDR imaging may improve indoor automated 3D reconstruction processes based on SfM methods. Verhoeven *et al.* [20] investigated the use of different grayscale conversion algorithms to decolorize color images as input for SfM software packages. Bellavia *et al.* [21] presented an online pre-processing strategy to detect and discard bad frames in video sequences. The method is based on the Double Window Adaptive Frame Selection (DWAFS) algorithm which works on a simple gradient statistic (gradient magnitude distribution). The percentile statistic of each frame is used to develop an adaptive decision strategy based on a dangling sample window according to the time series of the ongoing percentile values and the last best ones.

2. Color Balance and Exposure Equalization

Color balance is the global adjustment of the intensity of the (red, green, and blue) colors in order to render them correctly. Color balance and exposure equalization is a key step to ensure: (i) faithful color appearance of a digitized artifact; and (ii) consistency of the color-to-gray conversion. This latter one (see Section 4) is a fundamental step as all feature extraction and image matching algorithms works using only the luminance channel. A correct color balance allows minimizing the typical problem of incorrectly detected areas (e.g., different luminance value for the same color and/or isoluminant colors) that strongly appear in case of surfaces of the same color or colors with the same luminance value. Therefore, the color balance procedure aims to produce radiometrically-calibrated images ensuring the consistency of surface colors in all the images (*i.e.*, as much as possible similar RGB values for homologous pixels). Starting from captured RAW images, our workflow includes (Section 2.2): exposure compensation, optical correction, sharpen, and color balance.

2.1. Color Spaces

The use of an appropriate color space to work and render images on screen is fundamental.

sRGB, a standard RGB color space created cooperatively by HP and Microsoft in 1996, is certainly the best choice as output color space for textures and to display rendered 3D models for several reasons including:

- sRGB is the default color space for HTML, CSS, SMIL and other web standards;
- sRGB is consistent among different monitors or video-projectors; and
- sRGB is implemented in the OpenGL graphic libraries, used in many rendering software.

However, the sRGB color space is very narrow and may produce loss of information, mainly in the acquisition and processing phases. To avoid these problems, a broader rendered color space is used, such as the Adobe-RGB (1998), which represents an excellent compromise between the amount of colors that can be codified and the possibility of displaying them on the screen. The use of Adobe-RGB (1998) allows avoiding possible inaccuracies of the sRGB color space in shadows (~25% luminance) as well as highlights (~75% luminance). Adobe-RGB (1998) expands its advantages to areas of intense orange, yellow and magenta color. As the sRGB is a “*de facto*” standard for consumer cameras storing images in JPEG format, it is advisable to use the RAW format, which normally map to a rendered color space as the Adobe-RGB or sRGB color space.

2.2. Proposed Approach

Between the two general approaches (color characterization *vs.* spectral sensitivities based on color targets) [22] we adopted this last technique that uses a set of differently colored samples measured with a spectrophotometer.

The most precise characterization for any given camera requires recording its output for all possible *stimuli* and comparing it with separately measured values for the same *stimuli* [23]. However, storage of such a quantity of data is impractical, and, therefore, the response of the device is captured for only a limited set of *stimuli*—normally for the acquisition conditions. The responses to these representative *stimuli* can then be used to calibrate the device for input *stimuli* that were not measured, finding the transformation between measured CIExyz values and stored RGB values. To find this transformation, several techniques have been developed, including look-up tables [24].

The method for evaluating and expressing color accuracy (“color characterization”) includes a physical reference chart acquired under standard conditions, a reference chart color space with the ideal data values for the chart, a way to relate or convert the device color space to the reference chart color space and, finally, a way to measure and show errors in the device’s rendering of the reference chart. The target GretagMacbeth ColourChecker [25] is employed during the image acquisitions, considering the measurements of each patch as reported in Pascale [26].

A captured color image containing the GretagMacbeth ColourChecker is neutralized, balanced and properly exposed. Using in-house software, an ICC (International Color Consortium) profile—assigned together with the Adobe-RGB (1998) color space of the RAW image—is generated. Before creating ICC profiles, a standard gamma correction ($\gamma = 2.2$) is applied, converting all images to the camera’s native linear color space, thus improving the quality of the profiles. A protocol is developed to use the same calibration for groups of images with the same features (*i.e.*, orientation, exposure and framed surfaces) thus to maintain consistency in the process and results.

The color accuracy is computed in terms of the mean camera chroma relative to the mean ideal chroma in the CIE color metric (ΔE^*_{00}) as defined in 2000 by CIE [27]:

$$\Delta E^*_{00} = \sqrt{\left(\frac{\Delta L'}{k_L S_L}\right)^2 + \left(\frac{\Delta C'}{k_C S_C}\right)^2 + \left(\frac{\Delta H'}{k_H S_H}\right)^2 + R_T \frac{\Delta C'}{k_C S_C} \frac{\Delta H'}{k_H S_H}} \quad (1)$$

This formula is a new version of the original one (1976) and is more suitable for our uses. It takes into consideration the problem of non-perceptual uniformity of the colors for which ΔE^*_{00} varies the weight of L^* depending on where the brightness range falls. Song and Luo [28] showed that the perceptible and acceptable color differences in complex images presented on a CRT (Cathode Ray Tube) monitor are approximately 2.2 and 4.5, respectively. In our case, the latter value was used as a strict reference for accuracy, defined from perception tests on the results obtained using this value.

Exposure error in f-stops was also evaluated on the plane of the target assumed as one the main object captured in the image. The ΔE^*_{00} and the exposure error calculations was computed using Imatest Studio software version 3.9.

From an operational point of view, the preservation of color fidelity throughout the image processing is ensured by:

- taking pictures in the most homogeneous operative conditions (aperture/exposure direction and intensity of light);
- including ColourChecker target inside the photographed scenes in order to correct the image radiometry;
- storing photos in RAW format; and
- using an appropriate color space from the beginning of the image processing.

An important and critical issue is the acquisition of the color target. In order to maintain uniform lighting in an external environment, for each image, we need to consider: (i) surfaces illuminated and

oriented as the ColourChecker and that presents an angle of incidence with sunlight of approximately 20° – 45° or (ii) image acquisitions performed with overcast sky. To minimize the light glare, that would give unexpected results in the calibration process, the ColourChecker is normally placed on a tripod with a dark background and orthogonal to the camera optical axis. Finally, we verified that a ColourChecker image width of 500 to 1500 pixels is sufficient for ΔE^*_{00} analysis, as also suggested in the Imatest user guide.

3. Image Denoising

Image noise is defined in the ISO 15739 standard as “unwanted variations in the response of an imaging system” [29]. It is formed when incoming light is converted from photons to an electrical signal and originates from the camera sensor, its sensitivity and the exposure time as well as by digital processing (or all these factors together). Noise can appear in different ways:

- *Fixed pattern noise* (“hot” and “cold” pixels): It is due to sensor defects or long time exposure, especially with high temperatures. Fixed pattern noise always appears in the same position.
- *Random noise*: It includes intensity and color fluctuations above and below the actual image intensity. They are always random at any exposure and more influenced by ISO speed.
- *Banding noise*: It is caused by unstable voltage power and is characterized by the straight band in frequency on the image. It is highly camera-dependent and more visible at high ISO speed and in dark image. Brightening the image or white balancing can increase the problem.
- *Luminance noise* (i.e., a variation in brightness): It is composed of noisy bright pixels that give the image a grainy appearance. High-frequency noise is prevalent in the luminance channel, which can range from fine grain to more distinct speckle noise. This type of noise does not significantly affect the image quality and can be left untreated or only minimally treated if needed.
- *Chrominance noise* (i.e., a variation in hue): It appears as clusters of colored pixels, usually green and magenta. It occurs when the luminance is low due to the inability of the sensor to differentiate color in low light levels. As a result, errors in the way color is recorded are visible and hence the appearance of color artifacts in the de-mosaicked image.

Starting from these considerations, the noise model can be approximated with two components:

- (a) A signal-independent Gaussian noise to compensate for the fixed pattern noise (FPN).
- (b) A signal-dependent Poisson noise to compensate for the temporal (random) noise, called Shot Noise.

A denoise processing basically attempts to eliminate—or at least minimize—these two components.

Several denoising methods [30–32] deal directly with Poisson noise. Wavelet-based denoising methods [33,34] adapt the transform threshold to the local noise level of the Poisson process. Recent papers on the Anscombe transform by Makitalo and Foi [35] and Foi [36], argue that, when combined with suitable forward and inverse variance-stabilizing transformations (VST), algorithms designed for homoscedastic Gaussian noise work just as well as *ad-hoc* algorithms based on signal-dependent noise models. This explains why the noise is assumed to be uniform, white and Gaussian, having previously applied a VST to the noisy image to take into account the Poisson component.

An effective restoration of image signals will require methods that either model the signal a-priori (i.e., Bayesian) or learn the underlying characteristics of the signal from the given data (i.e., learning, non-parametric, or empirical Bayes’ methods). Most recently, the latter approach has become very popular, mainly using patch-based methods that exploit both local and non-local redundancies and “self-similarities” in the images [24]. A patch-based algorithm denoises each pixel by using knowledge of (a) the patch surrounding it and (b) the probability density of all existing patches.

Typical noise reduction software reduces the visibility of noise by smoothing the image, while preserving its details. The classic methods estimate white homoscedastic noise only, but they can be adapted easily to estimate signal- and scale-dependent noise.

The main goals of image denoising algorithms are:

- perceptually flat regions should be as smooth as possible and noise should be completely removed from these regions;
- image boundaries should be well preserved and not blurred;
- texture detail should not be lost;
- the global contrast should be preserved (*i.e.*, the low-frequencies of denoised and input images should be equal); and
- no artifacts should appear in the denoised image.

All these goals are appropriate also for our cases where we need not to remove signal, nor distort blob shape and intensity areas to have efficient keypoint extraction and image matching processing.

Numerous methods were developed to meet these goals, but they all rely on the same basic method to eliminate noise: averaging. The concept of averaging is simple, but determining which pixels to average is not.

In summary:

- the noise model is different for each image;
- the noise is signal-dependent;
- the noise is scale-dependent; and
- the knowledge of each dependence is crucial to proper denoising of any given image which is not raw, and for which the camera model is available.

To meet this challenge, four denoising principles are normally considered:

- transform thresholding (sparsity of patches in a fixed basis);
- sparse coding (sparsity on a learned dictionary);
- pixel averaging and block averaging (image self-similarity); and
- Bayesian patch-based methods (Gaussian patch model).

Each principle implies a model for the ideal noiseless image. The current state-of-the-art denoising recipes are in fact a smart combination of all these ingredients.

3.1. Evaluated Methods

We investigated different denoise algorithms, some commercial, namely:

- *Imagenomic Noiseware* [37,38]: It uses hierarchical noise reduction algorithms, subdividing the image noise into two categories: luminance noise and color noise, furthermore divided into frequencies ranging from very low to high. The method includes detection of edges and processing at different spatial frequencies, using the YCbCr color space. Noiseware presents good quality results and it is easy to set-up.
- *Adobe Camera RAW* denoise [39]: Noise-reduction available in Camera Raw 6 with Process 2010 (in Section 6 simply called *Adobe*) uses a luminance noise-reduction technique based on a wavelet algorithm that seeks to determine extremely high-frequency noise and to separate it from high-frequency image texture. The method is capable of denoising large noisy areas of an image as well as to find and fix “outliers”, *i.e.*, localized noisy areas. Unfortunately, the method is a global filter and it needs a skilled manual intervention for each image to set-up the right parameters.
- *Non-Local Bayesian* filter [40–42]: It is an improved patch-based variant of the Non Local-means (NL-means) algorithm, a relatively simple generalization of the Bilateral Filter. In the NL-Bayes algorithm, each patch is replaced by a weighted mean of the most similar patches present in a neighborhood. To each patch is associated a mean (which would be the result of NL-means), but also a covariance matrix estimating the variability of the patch group. This allows computing an optimal (Bayesian minimal mean square error) estimate of each noisy patch in the group, by a

simple matrix inversion. The implementation proceeds in two identical iterations, but the second iteration uses the denoised image of the first iteration to estimate better the mean and covariance of the patch models.

- *Noise Clinic* [43–46]: It is the conjunction of a noise estimation method and of a denoising method. Noise estimation is with an extension of [47] method to be able to estimate signal-dependent noise, followed by multiscale NL-Bayes denoising method. The multiscale denoising follow these principles: (a) signal dependent noise estimated at each scale; and (b) zoom down followed by Anscombe transform to whiten the noise at each scale; denoising performed at each scale, bottom-up (coarse to fine). Noise Clinic is implemented in DxO Optics Pro with the name of Prime (Probabilistic Raw IMage Enhancement), and it is useful for very noisy and high-ISO RAW images, or for photos taken with an old camera that could not shoot good-quality images at ISO higher than 1600 ISO.
- *Color Block Matching 3D (CBM3D)* filter [48]: A color variant of Block Matching 3D (BM3D) filter [49]. BM3D is a sliding-window denoising method extending the Discrete Cosine Transform (DCT) [25] and NL-means algorithms. BM3D, instead of adapting locally a basis or choosing from a large dictionary, uses a fixed basis. The main difference from DCT denoising is that a set of similar patches is used to form a 3D block, which is filtered by using a 3D transform, hence the name “collaborative filtering”. The algorithm works in two stages: “basic estimate” of the image and the creation of the final image, and with four steps each stage: (a) finding the image patches similar to a given image patch and grouping them in a three-dimensional block; (b) 3D linear transform of the 3D block; (c) shrinkage of the transform spectrum coefficients; and (d) inverse three-dimensional transformation. This second step mimics the first step, with two differences. The first difference is that it compares the filtered patches instead of the original patches. The second difference is that the new 3D group is processed by an oracle Wiener filter, using coefficients from the denoised image obtained at the first step to approximate the true coefficients. The final aggregation step is identical to that of the first step. CBM3D extends the multi-stage approach of BM3D via the YoUoVo color system. CBM3D produces a basic estimate of the image, using the luminance data, and delivers the denoised image performing a second stage on each of the three color channels separately. This generalization of the BM3D is non-trivial because authors do not apply the grayscale BM3D independently on the three luminance-chrominance channels but they impose a grouping constraint on both chrominance. The grouping constraint means that the grouping is done only once, in the luminance (which typically has a higher SNR than the chrominance), and exactly the same grouping is reused for filtering both chrominance. The constraint on the chrominance increases the stability of the grouping with respect to noise. With this solution, the quality of denoised images is also excellent for moderate noise levels.

3.2. Proposed Approach

Following the experiment results, an in-house method (named *CBM3D-new*) was developed starting from the CBM3D approach [48]. For every image of a dataset, the method automatically select the necessary parameters based on the type of camera, ISO sensitivity and stored color profiles. In particular, the processing selection of the latter one is based on image features and camera capabilities: dealing with professional or prosumer setups, when source images are stored as RAW images or in non-RAW formats characterized by a wide color space such as the Adobe-RGB (1998), then opponent color space are chosen. When source images are stored in JPG format using a relatively narrower color space, such as sRGB—the most used in consumer cameras—then YCbCr color space is chosen.

The camera ISO is strictly related to the image noise. The sigma parameter, *i.e.*, the standard deviation of the noise, increases when the ISO increases, ranging from lower values ($\sigma = 1$ for images shot at less than 100 ISO) to higher ones ($\sigma = 10$ for images shot at more than 800 ISO). ISO sensitivity similarly influences other filtering parameters, such as the number of sliding step to process every

image block (ranging from 3 to 6), the length of the side of the search neighborhood for full-search block-matching (ranging from 25 to 39) as well as the number of step forcing to switch to neighborhood full-search (ranging from 1 to 36).

4. Color-to-Gray

Most of the algorithms involved in the image-based 3D reconstruction pipeline (mainly feature extraction for tie points identification and dense image matching) are conceptually designed to work on grayscale images (*i.e.*, single-band images) instead of the RGB triple. This is basically done to highly reduce the computational complexity of the algorithms compared to the utilization of the three channels. Color to grayscale conversion (or decolorization) can be seen as a dimensionality reduction problem and it should not be underestimated, as there are many different properties that need to be preserved. Over the past decades, different color-to-gray algorithms have been developed to derive the best possible decolorized version of a color image [20]. All of them focus on the reproduction of color images with grayscale mediums, with the goal of: (i) a perceptual accuracy in terms of the fidelity of the converted image; and (ii) a preservation of the color contrast and image structure contained in the original color also in the final decolorized image. Nevertheless, these kinds of approaches are not designed to fulfill the needs of image matching algorithms where local contrast preservation is crucial during the matching process. This was also observed in Lowe [50] where the candidate key points with low contrast are rejected in order to decrease the ambiguity of the matching process.

Color-to-gray conversion methods can be classified according to their working space:

- *Color Space* (linear or non-linear): The CIE Y method is a widely used conversion, based on the CIE 1931 XYZ color space. It takes the XYZ representation of the image and uses Y as the grayscale value.
- *Image Space* (also called functional): Following Benedetti *et al.* [51], these methods can be divided in three groups.
 - (a) *Trivial methods*: They are the most basic and simple ones, as they do not take into account the spectral power distribution (SPD) of the color channels. They lose a lot of image information as for every pixel they discard two of the three color values, or discard one value averaging the remaining ones, not taking into account any color properties. Despite this loss, they are commonly used for their simplicity and speed. A typical trivial method is the RGB Channel Filter that selects a channel between R, G or B and uses this channel as the grayscale value (this method is afterwards called *GREEN2GRAY*).
 - (b) *Direct methods*: The conversion is a linear function of the pixel's color values. Typically, this class of functions takes into account the spectrum of different colors. The Naive Mean direct method takes the mean of the color channels. With respect to trivial methods, it takes information from every channel, though it does not consider the relative spectral power distribution (SPD) of the RGB channels. The most popular of these methods is *RGB2GRAY* that uses the NTSC CCIR (National Television System Committee—Consultative Committee on International Radio) 601 luma weights, with the formula:

$$Y' = 0.2989R' + 0.5870G' + 0.1140B' \quad (2)$$

Other weights can be used, according to the users and software (e.g., Adobe Photoshop uses these specific weights for the channels R, G, and B: 0.4, 0.4, 0.2).

- (c) *Chrominance direct methods*: They are based on more advanced algorithms, trying to mitigate the problem related to isoluminant colors. They assign different grayscale values to isoluminant colors, altering the luminance information and using the chrominance information. In order to increase or decrease the “correct” luminance to differentiate

isoluminant colors, these methods exploit a result from studies on human color perception, known as the Helmholtz–Kohlrausch (H-K) effect [52]. The H-K effect states that the perceived lightness of a stimulus changes as a function of the chroma. This phenomenon is predicted by a chromatic lightness term that corrects the luminance based on the color's chromatic component and on starting color space. Chrominance direct methods can be performed either locally [53,54] or globally [55]. Local methods make pixels in the color image not processed in the same way and usually rely on the local chrominance edges for enhancement. Global methods strive to produce one mapping function for the whole image thus producing same luminance for the same RGB triplets and high-speed conversion.

4.1. Evaluated Methods

We investigated different color-to-gray methods, namely:

- *GREEN2GRAY*: It is a trivial method working in Image Space where the green channel is extracted from a RGB image and used to create the final grayscale image.
- *Matlab RGB2GRAY*: It is a direct method implemented in Matlab and based on the above mentioned weighted sum of the three separate channels.
- *Decolorize* [55]: The technique performs a global grayscale conversion by expressing the grayscale as a continuous, image-dependent, piecewise linear mapping of the primary RGB colors and their saturation. Their algorithm works in the YPQ color opponent space and aims to perform a contrast enhancement too. The color differences in this color space are projected onto the two predominant chromatic contrast axes and are then added to the luminance image. Unlike a principal component analysis, which optimizes the variability of observations, a predominant component analysis optimizes the differences between observations. The predominant chromatic axis aims to capture, with a single chromatic coordinate, the color contrast information that is lost in the luminance channel. The luminance channel Y is obtained with the NTSC CCIR 601 luma weights. The method is very sensitive to the issue of gamma compression with some risks of decrease of the quality of the results mainly in light areas or dark areas where many features will be lost because the saturation balancing interacts incorrectly with the outlier detection.
- *Realtime* [56–58]: This method is based on the consideration that in the human visual system the relationship to the adjacent context plays a vital role to order the different colors. Therefore, the method relaxes the color order constraint and seeks better preservation of color contrast and significant enhancement of visual distinctiveness for edges. For color pairs without a clear order in brightness, a bimodal distribution (*i.e.*, a mixture of two Gaussians) is performed to automatically find suitable orders with respect to the visual context in optimization. This strategy enables automatically finding suitable grayscales and preserves significant color changes. Practically the method uses a global mapping scheme where all color pixels in the input are converted to grayscale using the same mapping function (a finite multivariate polynomial function). Therefore, two pixels with the same color will have the same grayscale. The technique is today implemented in OpenCV 3.0. In order to achieve real-time performance, a discrete searching optimization can be used.
- *Adobe Photoshop*.

To evaluate the performances of the aforementioned methods (Figure 5), we applied the pixel-by-pixel difference method applying an offset of 127 levels of brightness to better identify the differences. This technique is the most appropriate to evaluate a method's efficiency for machine readable process. The simple image subtraction can rapidly provide visual results rather than using CIELAB ΔE^*ab or other perceptually-based image comparison methods.

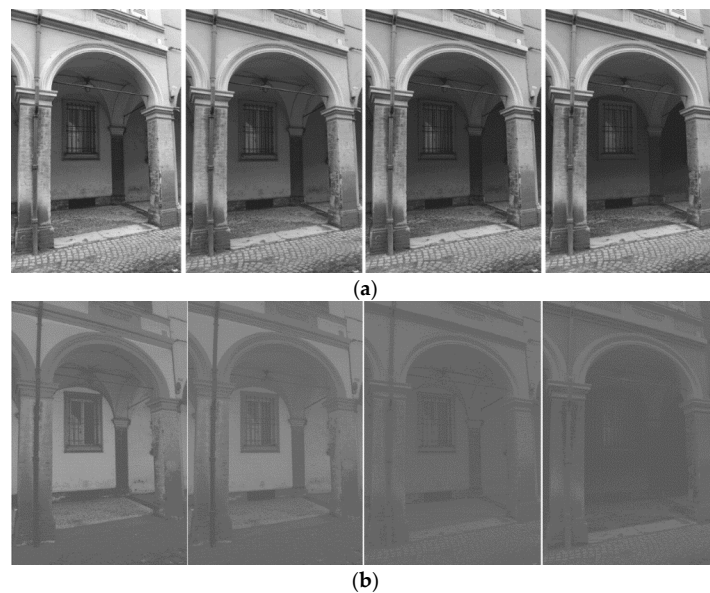


Figure 5. Results of color-to-gray conversion methods: produced gray scale images (a) and image difference (b) between (from left to right) *Adobe Photoshop* and *Decolourize*, *Adobe Photoshop* and *Realtime*, *Adobe Photoshop* and *Matlab RGB2GRAY*, and *Adobe Photoshop* and *BID*.

4.2. Proposed Approach

Based on the results achieved with the aforementioned methods, a new decolorization technique, named *Bruteforce Isoluminants Decrease (BID)*, was developed. The aim of *BID* is to preserve the consistency between different images considering the following requirements.

- *Feature discriminability*: The decolorization method should preserve the image features discriminability in order to match them in as many images as possible.
- *Chrominance awareness*: The method should distinguish between isoluminant colors.
- *Global mapping*: While the algorithm can use spatial information to determine the mapping, the same color should be mapped to the same grayscale value for every pixel in the image.
- *Color consistency*: The same color should be mapped to the same grayscale value in every image of the dataset.
- *Grayscale preservation*: If a pixel in the color image is already achromatic, it should maintain the same gray level in the grayscale image.
- *Unsupervised algorithm*: It should not need user tuning to work properly, in particular for large datasets.

BID computes the statistical properties of the input dataset with the help of a representative collection of image patches. Differently from the *Multi-Image Decolourize* method [51], *BID* is a generalization of the *Matlab RGB2GRAY* algorithm, which simultaneously takes in input and analyses the whole set of images that need to be decolorized. *BID* has its foundation in the statistics of extreme-value distributions of the considered images and presents a more flexible strategy, adapting dynamically channel weights depending on specific input images, in order to find the most appropriate weights for a given color image. *BID* preserves as much as possible the amount of the conveyed information. The algorithm behind *BID* tries to maximize the number of peaks obtained in the converted image and to distribute as evenly as possible the amount of tones by evaluating the goodness of a fitting distribution. To calculate the best rectangular fitting, we assumed a 0 slope regression line. The general equation of the regression line is:

$$\beta = \bar{y} - m\bar{x} \quad (3)$$

where β is equivalent to the average of the histogram points. After calculating the average, the minimum error within all the calculated combinations of channel mixings is sought. The error is calculated as a least squares error:

$$S = \sum_{i=1}^n (y_i - \beta)^2 \quad (4)$$

where y_i are the actual points, while β is the best linear fitting of the histogram. *BID* cyclically varies the amount of red, green and blue and for each variation calculates the distribution of the resulting grayscale image and assesses the fitting quality with respect to a rectangular distribution. Then, *BID* chooses the mixing that maximizes the number of tones obtained in the converted image. Finally, similarly to Song *et al.* [59], *BID* uses a measurement criterion to evaluate the decolorization quality, *i.e.*, the newly defined dominant color hypothesis.

Figure 6 reports an example of *BID* results with respect to *Matlab RGB2GRAY* method. The main disadvantage of the developed method is the high computational pre-processing time due to the sampled patches on each image of the dataset.

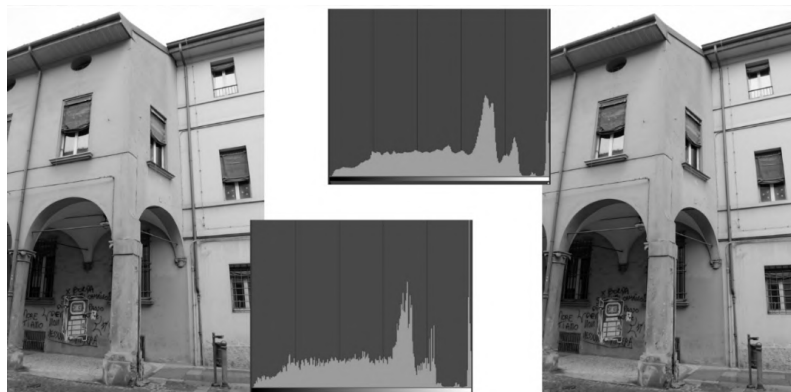


Figure 6. A comparison between *Matlab RGB2GRAY* (left) and *BID* (right) that shows a smoother histogram.

5. Image Content Enhancement with Wallis Filtering

Image contents play a fundamental role in many processing and feature extraction methods. There are various enhancement algorithms to sharp and increase the image quality [9,60–62]. For image-based 3D reconstruction purposes, low-texture surface (such as plaster building facades) causes difficulties to feature detection methods (such as the Difference-of-Gaussian (DoG) function) and matching algorithms, leading to outliers and unsuccessful matching results. Among the proposed methods to enhance image contents, the Wallis filter [11] showed very successful performances in the photogrammetric community [63–67]. Jazayeri *et al.* [68] tested the Wallis filter for different parameters to evaluate its performances for interest point detection and description. Those results demonstrated that an optimum range of values exists and depending on the requirements of the user, but automatic value selection remains undetermined.

The filter is a digital image processing function that enhances the contrast levels and flattens the different exposure to achieve similar brightness of gray level values. The filter uses two parameters to control the enhancement's amount, the *contrast expansion factor A* and the *brightness forcing factor B*. The algorithm is adaptive and adjusts pixel brightness values in local areas only, contrary to a global contrast filter, which applies the same level of contrast throughout an entire image. The resulting enhanced image contains greater detail in both low and high-level contrast regions concurrently, ensuring that good local enhancement is achieved throughout the entire image. The Wallis filter requires the user to accurately set a target mean and standard deviation in order to locally adjust areas and match the user-specified target values. Firstly, the filter divides the input image into neighboring

square blocks with a user-defined size (“window size”) in order to calculate local statistics. Then, mean (M) and standard deviation (S) of the unfiltered image are calculated for each individual block based on the gray values of the pixels and the resulting value is assigned to the central cell of each block. The mean and standard deviation values of all other cells in the block are calculated from this central cell by bilinear interpolation. In this way, each individual pixel gets its own initial local mean and standard deviation based on surrounding pixel values. The user-defined mean and standard deviation values are then used to adjust the brightness and the contrast of the input cells. The resulting enhanced image is thus a weighted combination of the original and user-defined mean and standard deviation of the image. The implementation of the filter of Wallis, given the aforementioned factor A and B , can be summarized as follows:

- let S be the standard deviation for the input image;
- let M be the mean for the input image;
- for each (x,y) pixel in the image,
- calculate local mean m and standard deviation s using a $N \times N$ neighborhood; and finally
- calculate the enhanced output image as

$$(x,y) = S \times (\text{input}(x,y) - m)/(s + A) + M * B + m \times (1 - B) \quad (5)$$

Characterization of Wallis Parameters

The quality of the Wallis filter procedure relies on two parameters: the *contrast expansion factor* A and the *brightness forcing factor* B . The main difficulty when using the Wallis filter is the correct selection of these parameters, in particular for large datasets, where a unique value of A or B could lead to unsuitable enhanced images. Although several authors reported parameters for successful projects, the filter is more an “*ad-hoc*” recipe than an easily deployable system for an automatic photogrammetric pipelines. To overcome this problem and following the achievement presented in [69], a Wallis parameters characterization study was carried out to automatically determine them. Three different datasets, each one composed of three images and involving the majority of possible surveying case studies were used (Figure 7):

1. a cross vault characterized by smooth bright-colored plaster;
2. a building facade and porticoes with smooth plaster; and
3. a Venetian floor, with asphalt and porphyry cubes with a background facade overexposed and an octagonal pillar in the foreground coated with smooth plaster.



Figure 7. Employed datasets (cross vault with plaster, building façade and Venetian floor, respectively) to study the automatic selection of the Wallis filter parameters.

For every dataset, the images were enhanced using different Wallis parameters and then matched to find homologues points using a calibrated version [7] of the SIFT operator available in Vedaldi's implementation [69]. This characterization procedure delivered the following considerations.

1. The number of extracted tie points is inversely proportional to the value of the parameter A , but the number of correct matches remains basically stable when varying A , which can then be set at high values to speed up the computation (6–8).
2. Varying the user-specified standard deviation, the number of tie points and correct matches increases substantially linearly up to a value of 100 and then remains constant (Figure 8a).

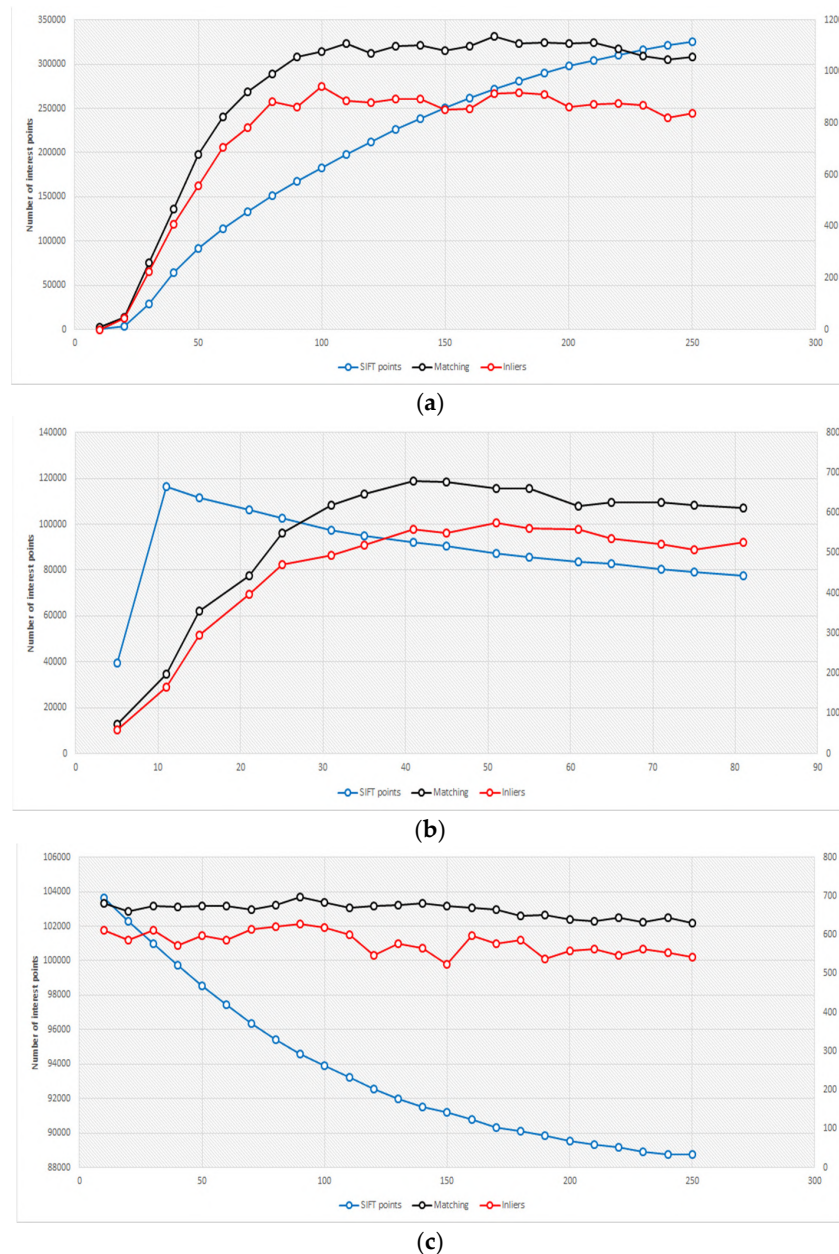


Figure 8. Results of the Wallis evaluation on the cross vault dataset. The charts report the number of extracted interest points with respect to the considered parameter values: (a) cross vault dataset—performance evaluation varying the standard deviation; (b) cross vault dataset—performance evaluation varying the window size; and (c) cross vault dataset—performance evaluation varying the mean.

3. Sensor resolution and window size are linearly related and the increasing of the window size beyond the optimal value does not involve any improvement in either the number of positive matches and in the number of extracted tie points (Figure 8b).
4. The mean presents optimal values between 100 and 150 with a decay afterwards (Figure 8c).

Starting from these observations, a new implementation of the Wallis filter was realized to select the optimal filter parameters and achieve the highest possible ratio of corrected matches with respect to the number of extracted tie points. In particular, the window size parameter is chosen according to the sensor resolution, and presents a linear variation starting from the experimental best values of 41 for a 14 MPixel sensor and 24 for a 10 MPixel sensor. According to our experimental trials, the standard deviation was forcefully set to 60 and the mean to 127. The Contrast Expansion Constant parameter (A) was set to 0.8 to increase the number of detected interest points located in homogeneous and texture-less areas and, alongside, to speed up the computation. The brightness forcing factor (B) according to the experimental results, was set to 1 if the image mean was lower than 150, linearly decreased otherwise, evaluating the entropy of the image.

Figure 9 shows the results of Wallis filtering: lower image contents are boosted, whereas a better histogram is achieved.



Figure 9. Example of the Wallis filtering procedure: note the much smoother and Gaussian-like histogram.

6. Assessment of the Proposed Methodology

The implemented pre-processing procedure was evaluated on various image networks featuring different imaging configurations, textureless areas and repeated pattern/features. The employed datasets try to verify the efficiency of different techniques in different situations (scale variation, camera rotation, affine transformations, *etc.*). The datasets contain convergent images, some orthogonal camera rolls and a variety of situations emblematic of failure cases, *i.e.*, 3D scenes (non-coplanar) with homogeneous regions, distinctive edge boundaries (e.g., buildings, windows, doors, cornices, arcades), repeated patterns (recurrent architectural elements, bricks, *etc.*), textureless surfaces and illumination changes. With respect to other evaluations where synthetic datasets, indoor scenarios, low resolution images, flat objects or simple two-view matching procedures are used and tested, such datasets are more varied with the aim of a complete and precise scene's 3D reconstruction.

All algorithms are tested and applied to raw images, *i.e.*, images as close as possible to the direct camera output retaining only the basic in-camera processing: black point subtraction, bad pixel

removal, dark frame, bias subtraction and flat-field correction, green channel equilibrium correction, and Bayer interpolation. The datasets are processed with different image orientation software (Visual SFM, Eos Photomodeler and Agisoft Photoscan), trying to keep a uniform number of extracted key points and tie points. Then, dense point clouds are extracted with a unique tool (nFrames SURE). The performances of the pre-processing strategies are reported using:

- (i) pairwise matching efficiency *i.e.*, number of correct inlier matches after the RANSAC (RANdom SAmple Consensus) phase normalized with all putative correspondences (Section 6.1);
- (ii) the statistical output of the bundle adjustment (Sections 6.2 and 6.3);
- (iii) the number and density of points in the dense point cloud (Sections 6.2 and 6.3); and
- (iv) an accuracy evaluation of the dense matching results (Section 6.2).

6.1. Dataset 1

The first dataset (four images acquired with a Nikon D3100, sensor size 23.1×15.4 mm, 18 mm nominal focal length) shows part of Palazzo Albergati (Bologna, Italy) characterized by repeated brick walls, stone cornices and a flat facade. The camera was moved along the façade of the building, then tilted and rotated (Figure 10). This set of images is used to evaluate the denoise and color-to-gray techniques with respect to the tie points extraction procedure. The pairwise matching is assessed using three camera movements: (i) parallel with short baseline (a,b); (ii) rotation of *ca.* 90° (a–d); and (iii) tilt of *ca.* 45° (b,c).



Figure 10. The images used to assess the pairwise matching efficiency of the tie point extraction procedure: parallel acquisitions with short baseline (a,b); tilt of *ca.* 45° (c) and rotation of *ca.* 90° (d).

Table 1 reports the computed pairwise matching efficiency after applying the various denoising methods and color-to-gray techniques. The developed methods (*CBM3D-new* and *BID*) demonstrate a better efficiency in the tie point extraction.

Table 1. Efficiency of each denoise and color-to-gray technique for the image pairs of Dataset 1. *Adobe* method is Camera Raw 6 with Process 2010 (see Section 3.1).

	<i>Denoising</i>				<i>Color-to-Gray</i>		
	Parallel 00-01	Rotate 90° 00-03	Tilt 45° 01-02		Parallel 00-01	Rotate 90° 00-03	Tilt 45° 01-02
<i>Adobe</i>	0.982	0.812	0.5	<i>Adobe</i>	0.992	0.821	0.630
<i>CBM3D</i>	0.991	0.837	0.561	<i>REALTIME</i>	0.992	0.827	0.618
<i>NLBayes</i>	0.978	0.801	0.651	<i>Decolourize</i>	0.992	0.863	0.626
<i>Noiseclinic</i>	0.984	0.769	0.473	<i>RGB2GRAY</i>	0.980	0.690	0.329
<i>IMAGENOMIC</i>	0.980	0.690	0.329	<i>GREEN2GRAY</i>	0.992	0.786	0.640
<i>Nodenoise</i>	0.975	0.679	0.335	<i>BID</i>	0.993	0.825	0.676

6.2. Dataset 2

The second dataset (35 images acquired with a Nikon D3100, sensor size 23.1×15.4 mm, 18 mm nominal focal length) concerns two spans of a three floors building (6×11 m) characterized by some arches, pillars, cross vaults and plastered walls with uniform texture. The camera was moved along the porticoes, with some closer shots of the columns (Figure 11). With this dataset we report how color balancing and denoising methodologies help improving the bundle adjustment and dense matching procedures. The accuracy evaluation of the dense matching results is done using a Terrestrial Laser Scanning (TLS) survey as reference (a Faro Focus3D was employed). Three regions (Figure 12A1–A3) are identified and compared with the photogrammetric dense clouds. The average image GSD (Ground Sample Distance) in the three regions of interest is *ca.* 2 mm but the dense matching was carried out using the second-level image pyramid, *i.e.*, at a quarter of the original image resolution. Therefore, in order to have a reference comparable to the dense matching results, the range data are subsampled to a grid of 5×5 mm.



Figure 11. Dataset 2 (35 images)—two arches of a portico.

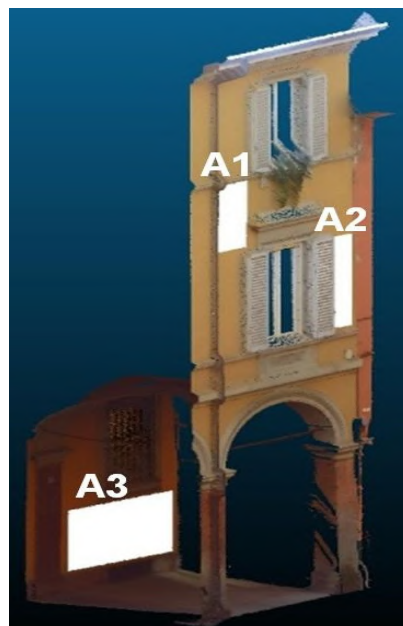


Figure 12. The three regions of interest (A1–A3) surveyed with TLS as ground truth.

6.2.1. Color Balance Results

The results of the orientation and dense matching steps are reported in Table 2. The color balancing procedure generally helps in increasing the number of oriented images, except with PS where the dataset is entirely oriented at every run. Furthermore, it helps in deriving denser point clouds.

Table 2. Bundle adjustment (BA) results and dense matching improvement on Dataset 2.

	<i>Not Enhanced</i>	<i>Color Balanced</i>
VisualSFM (VSFM)		
Numb. oriented images	31	33
BA quality (px)	0.48	0.48
EOS Photomodeler (PM)		
Numb. oriented images	31	33
BA reprojection error (px)	1.44	0.89
Agisoft Photoscan (PS)		
Numb. oriented images	35	35
BA reprojection error (px)	0.51	0.54
Dense Matching (SURE)		
# 3D points	1,259,795	1,626,267


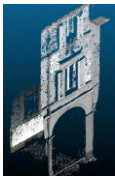
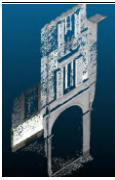


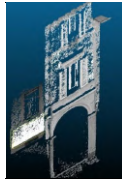
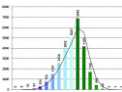
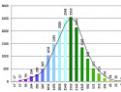
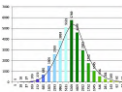
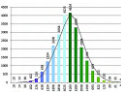
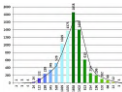
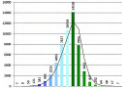
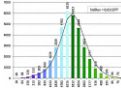
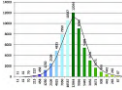
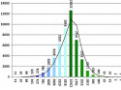
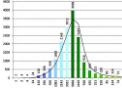
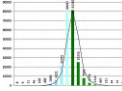
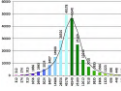
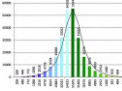
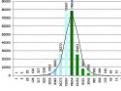
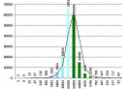
6.2.2. Image Denoising Results

The denoising methods are coupled to *Matlab RGB2GRAY* and Wallis filtering before running the automated orientation and matching procedures. The achieved adjustment results, according to the different denoising methods, show that more images can be oriented (Table 3) and denser point clouds can be retrieved (Table 4).

Table 3. Bundle adjustment (BA) results for every denoising procedure on Dataset 2. *Adobe* method is Camera Raw 6 with Process 2010 (see Section 3.1). On such a small dataset, Photoscan seems to be quite robust even without any pre-processing method.

	<i>No Denoise</i>	<i>Adobe</i>	<i>CBM3D-New</i>	<i>NL-Bayes</i>	<i>Noise Clinic</i>	<i>Image Nomic</i>
VisualSFM (VSFM)						
Numb. oriented images	32	35	35	35	35	35
BA quality (px)	0.29	0.48	0.48	0.45	0.69	0.37
EOS PhotoModeler (PM)						
Numb. oriented images	33	33	33	33	33	31
BA reprojection error (px)	0.89	0.84	0.83	0.86	0.87	0.88
Agisoft Photoscan (PS)						
Numb. oriented images	35	35	35	35	35	35
BA reprojection error (px)	0.49	0.49	0.49	0.49	0.53	0.55

Table 4. Evaluation of the denoising procedures on the dense matching phase. The ground truth is given by a TLS survey resampled to 5×5 mm grid. The histograms of the cloud-to-cloud point distribution errors are also reported.

		<i>No Denoise</i>	<i>Adobe</i>	<i>CBM3D-New</i>	<i>NL-Bayes</i>	<i>Noise Clinic</i>	<i>Image Nomic</i>
Numb. 3D points		998,995	1,308,768	1,456,024	1,456,561	1,428,996	1,346,559
							
A1 (0.8×1.4 m): 36.294 pts	# 3D points	1227	27,508	16,257	31,386	23,419	8194
	Std Dev (mm)	N/A	12.26	13.61	11.73	12.07	17.22
							
A2 (0.5×1.9 m): 33.868 pts	# 3D points	495	25,327	17,966	29,632	23,541	7548
	Std Dev (mm)	N/A	11.18	11.73	11.90	11.04	19.00
							
A3 (2.6×1.3 m): 120.222 pts	# 3D points	32,014	176,478	179,553	183,166	184,835	160,120
	Std Dev (mm)	N/A	11.76	12.35	11.44	10.03	10.99
							




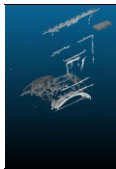

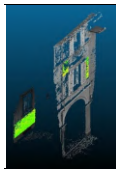
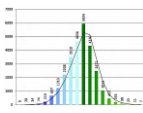
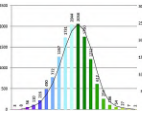
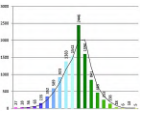
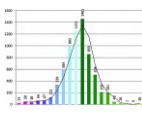
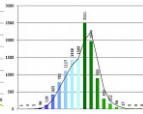
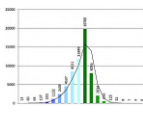
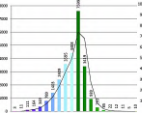
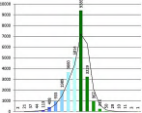
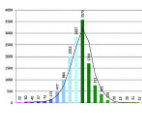
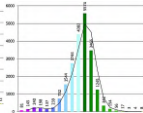
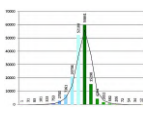
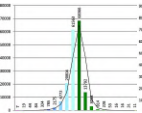
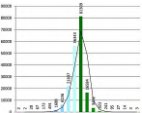
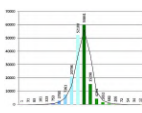
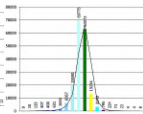
6.2.3. Color-to-Gray Results

The color-to-gray conversion (coupled with the Wallis filtering) shows how algorithms are differently affecting the BA procedure (Table 5) as well as the dense matching results (Table 6). It can be generally noticed that the proposed *BID* method allows retrieving a larger number of oriented images, a better re-projection errors and denser point clouds.

Table 5. Bundle adjustment (BA) results of the various color-to-gray procedures on Dataset 2.

	<i>GREEN2GRAY</i>	<i>Adobe</i>	<i>Realtime</i>	<i>Decolourize</i>	<i>RGB2GRAY</i>	<i>BID</i>
VisualSFM (VSFM)						
Numb. oriented images	35	33	30	16	33	35
BA reprojection error (px)	0.55	0.42	0.37	0.38	0.35	0.58
EOS PhotoModeler (PM)						
Numb. oriented images	35	33	33	28	32	35
BA reprojection error (px)	0.87	0.89	0.87	0.89	0.88	0.86
Agisoft Photoscan (PS)						
Numb. oriented images	35	35	35	35	35	35
BA reprojection error (px)	0.52	0.53	0.54	0.54	0.52	0.51

Table 6. Dense matching results of the various color-to-gray procedures on Dataset 2. Comparison with respect to the ground truth TLS data are also shown. The histograms represent the point distribution errors of the cloud-to-cloud alignments.

		<i>GREEN2GRAY</i>	<i>Adobe</i>	<i>Realtime</i>	<i>Decolourize</i>	<i>RGB2GRAY</i>	<i>BID</i>
Numb. 3D points		1,703,607	1,444,269	1,522,044	1,375,971	1,184,432	1,964,397
							
A1—36,294 pts	# 3D points	27,127	12,776	10,890	N/A	6863	11,360
	Std Dev (mm)	11.28	14.83	12.78	N/A	15.13	18.83
					N/A		
A2—33,868 pts	# 3D points	32,619	12,853	15,264	N/A	6520	9950
	Std Dev (mm)	10.76	13.82	10.77	N/A	12.39	22.29
					N/A		
A3—120,222 pts	# 3D points	162,563	153,281	163,246	N/A	152,923	157,837
	Std Dev (mm)	11.95	10.82	8.71	N/A	8.89	10.82
					N/A		

6.3. Dataset 3

The last dataset (265 images acquired with a Nikon D3100, sensor size 23.1×15.4 mm, 18 mm nominal focal length) regards a three floors historical building (19 m height \times 10 m width), characterized by arcades with four arches, columns, cross vaults and plastered walls with uniform texture. The camera was moved along the porticoes, with some closer shots of the columns and of the vaults (Figure 13). The dataset is used to show how the various steps of the proposed pre-processing pipeline positively influence the 3D reconstruction procedure. Table 7 reports the achieved image orientation results without and with pre-processing. The developed methods (*CBM3D-new* for the noise reduction and *BID* for the color-to-gray conversion) allow orienting larger numbers of images.



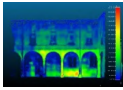
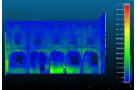
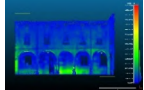
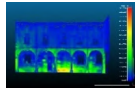
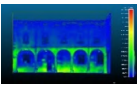
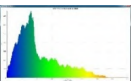
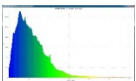
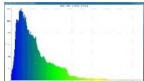
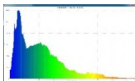
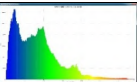
Figure 13. Dataset 3 (265 images)—a portico with arcades, columns, vaults and plastered walls.

Table 7. Result of the image orientation on Datasets 3 (265 images) without or with various pre-processing methods.

	<i>No Pre-Processing</i>	<i>No Color Correction, Only Wallis</i>	<i>Color Correction, No Denoise, RGB2GRAY, Wallis</i>	<i>CBM3D-New, RGB2GRAY, Wallis</i>	<i>CBM3D-New, BID, Wallis</i>
VisualSFM (VSFM)					
Numb. oriented images	90	218	214	232	249
	(2nd run)	(4th run)	(4th run)	(4th run)	(1st run)
BA quality (px)	0.739	0.650	0.285	0.411	0.750
PhotoModeler Scanner (PM)					
Numb. oriented images	55	N/A	180	247	247
BA quality (px)	0.687	N/A	0.779	0.804	0.790
Agisoft Photoscan (PS)					
Numb. oriented images	262	264	265	265	265
BA quality (px)	0.695	0.696	0.689	0.687	0.679

The dense matching procedure is then applied starting from the orientation results achieved in Photoscan results (Table 8). In this case, it is also shown how an appropriate pre-processing procedure allows deriving denser point clouds. The point density distribution for the different dense clouds (named Local Density Computation) was estimated. The density has been computed using a tool able to count, for each 3D point of the cloud, the number of neighbors N (inside a sphere of a radius R , fixed at 20 cm). The results of the Local Density Computation (shown as color-coded maps and histograms) show that the successive combination of the proposed pre-processing methods gradually achieve, beside an increasing amount of 3D points, a higher density and a more uniform distribution of points.

Table 8. Results of dense matching on Dataset 3 without and with various pre-processing methods. The Local Density Computation (radius: 20 cm) is also reported. The color-code maps show the matched point distribution on the surveyed scene. The histogram reports the number of neighbors (X-axis) *versus* the number of matched points in the cloud (Y-axis).

	No Pre-Processing	No Color Correction, Only Wallis	Color Correction, No Denoise, RGB2GRAY, Wallis	CBM3D-New, RGB2GRAY, Wallis	CBM3D-New, BID, Wallis
	Dense Matching (nFrames SURE)				
# 3D points	7,086,643	13,741,089	14,248,488	14,770,940	18,266,571
Maps of the number of neighbors					
Histograms of the number of neighbors					

7. Conclusions

The paper reported a pre-processing methodology to improve the results of the automated photogrammetric pipeline for 3D scene reconstruction. The developed pipeline consists of color balancing, image denoising, color-to-gray conversion and image content enrichment. Two new methods for image denoising (*CBM3D-new*) and grayscale reduction (*BID*) were also presented. The pipeline was evaluated using some datasets of architectural scenarios and advantages were reported. From the results in Section 6, it is clear that the pre-processing procedure, which requires very limited processing time, generally positively influences the performances of the orientation and dense matching algorithms. The evaluation shows how the combination of the various methods is indeed helping in achieving complete orientation results, sometimes better BA re-projection errors (although it is not a real measure of better quality) and, above all, denser and complete 3D dense point clouds.

As shown in the results, the best strategy implies applying a color enhancement, a denoise procedure based on the *CBM3D-new* method, the *BID* method for grayscale conversion and the Wallis filtering. This latter filtering seems to be fundamental also in the orientation procedure and not only when applying dense matching algorithms (as reported in the literature).

The developed pre-processing method is quite flexible and features the following characteristics.

- It is derived from many experiments and merging various state-of-the-art methods.
- The setting parameters can be fixed reading the image metadata (EXIF header).
- The denoising and grayscale conversion consider the entire dataset and they are not image-dependent.
- It is customized for improving the automated 3D reconstruction procedure.
- It is not based only on perceptual criteria typically used in image enhancement algorithms.
- It also gives advantages to the texture mapping phase.

The presented research substantially improved our 3D reconstruction pipeline and allows us to model large architectural scenarios for documentation, conservation and communication purposes (Figure 14).



Figure 14. Example of dense 3D point cloud of a portico obtained with the presented photogrammetric methodology based on an advanced pre-processing pipeline.

Author Contributions: All authors equally contributed to the procedure development, data processing and paper preparation.

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

References

1. Snavely, N.; Seitz, S.M.; Szeliski, R. Modeling the world from internet photo collections. *Int. J. Comput. Vis.* **2008**, *80*, 189–210. [[CrossRef](#)]
2. Barazzetti, L.; Scaioni, M.; Remondino, F. Orientation and 3D modeling from markerless terrestrial images: Combining accuracy with automation. *Photogramm. Rec.* **2010**, *25*, 356–381. [[CrossRef](#)]
3. Heinly, J.; Schönberger, J.L.; Dunn, E.; Frahm, J.M. Reconstructing the World* in Six Days*(As Captured by the Yahoo 100 Million Image Dataset). In Proceedings of the IEEE CVPR, Boston, MA, USA, 7–12 June 2015.
4. Pierrot-Deseilligny, M.; de Luca, L.; Remondino, F. Automated image-based procedures for accurate artifacts 3D modeling and orthoimage generation. *Geoinform. FCE CTU J.* **2011**, *6*, 291–299. [[CrossRef](#)]
5. Remondino, F.; Del Pizzo, S.; Kersten, T.P.; Troisi, S. Low-cost and open-source solutions for automated image orientation—A critical overview. In Progress in Cultural Heritage Preservation, Proceedings of the 4th International Conference, EuroMed 2012, Lemessos, Cyprus, 29 October–3 November 2012; pp. 40–54.
6. Crandall, D.; Owens, A.; Snavely, N.; Huttenlocher, D. SfM with MRFs: Discrete-Continuous Optimization for Large-Scale Structure from Motion. *IEEE PAMI* **2013**, *35*, 2841–2853. [[CrossRef](#)] [[PubMed](#)]
7. Apollonio, F.I.; Ballabeni, A.; Gaiani, M.; Remondino, F. Evaluation of feature-based methods for automated network orientation. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2014**, *XL-5*, 47–54. [[CrossRef](#)]
8. Remondino, F.; Spera, M.G.; Nocerino, E.; Menna, F.; Nex, F. State of the art in high density image matching. *Photogramm. Rec.* **2014**, *29*, 144–166. [[CrossRef](#)]
9. Gonzalez, R.F.; Woods, R. *Digital Image Preprocessing*; Prentice Hall: Upper Saddle River, NJ, USA, 2007; p. 976.

10. Milanfar, P. A tour of modern image filtering: New insights and methods, both practical and theoretical. *Signal Process. Mag.* **2013**, *30*, 106–128. [[CrossRef](#)]
11. Wallis, R. An approach to the space variant restoration and enhancement of images. In Proceedings of the Symposium on Current Mathematical Problems in Image Science, Monterey, CA, USA, 10–12 November 1976; pp. 329–340.
12. Degenhard, A.; Tanner, C.; Hayes, C.; Hawkes, D.J.; Leach, M.O. Pre-processed image reconstruction applied to breast and brain MR imaging. *Physiol. Meas.* **2001**, *22*, 589–604. [[CrossRef](#)] [[PubMed](#)]
13. Mahiddine, A.; Seinturier, J.; Boi, D.P.J.; Drap, P.; Merad, D.; Long, L. Underwater image preprocessing for automated photogrammetry in high turbidity water: An application on the Arles-Rhone XIII roman wreck in the Rhodano River, France. In Proceedings of the IEEE VISM Conference, Milan, Italy, 2–5 September 2012; pp. 189–194.
14. Guidi, G.; Gonizzi, S.; Micoli, L.L. Image pre-processing for optimizing automated photogrammetry performances. *ISPRS Int. Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2014**, *II-5*, 145–152. [[CrossRef](#)]
15. Maini, R.; Aggarwal, H. A comprehensive review of image enhancement techniques. *J. Comput.* **2010**, *2*, 8–13.
16. Stamos, I.; Liu, L.; Chen, C.; Wolberg, G.; Yu, G.; Zokai, S. Integrating automated range registration with multiview geometry for the photorealistic modeling of large-scale scenes. *Int. J. Comput. Vis.* **2008**, *78*, 237–260. [[CrossRef](#)]
17. Klein, G.; Murray, D. Improving the agility of keyframe-based SLAM. In Proceedings of the 10th ECCV Conference, Marseille, France, 12–18 October 2008; pp. 802–815.
18. Mei, C.; Reid, I. Modeling and generating complex motion blur for real-time tracking. In Proceedings of the IEEE CVPR, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
19. Lee, H.S.; Kwon, J.; Lee, K.M. Simultaneous localization, mapping and deblurring. In Proceedings of the IEEE ICCV Conference, Barcelona, Spain, 6–13 November 2011; pp. 1203–1210.
20. Verhoeven, G.; Karel, W.; Štuhec, S.; Doneus, M.; Trinks, I.; Pfeifer, N. Mind your gray tones—Examining the influence of decolourization methods on interest point extraction and matching for architectural image-based modelling. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2015**, *XL-5/W4*, 307–314. [[CrossRef](#)]
21. Bellavia, F.; Fanfani, M.; Colombo, C. Fast Adaptive Frame Preprocessing for 3D Reconstruction. In Proceedings of the VISAPP Conference, Berlin, Germany, 11–14 March 2015; pp. 260–267.
22. Hong, G.; Luo, M.R.; Rhodes, P.A. A study of digital camera colourimetric characterization based on polynomial modeling. *Colour Res. Appl.* **2001**, *26*, 76–84. [[CrossRef](#)]
23. Yaroslavsky, L.P. Local adaptive image restoration and enhancement with the use of DFT and DCT in a running window. *Proc. SPIE* **1996**, *2825*, 2–13.
24. Reinhard, E.; Arif Khan, E.; Oguz Akyüz, A.; Johnson, G. *Colour Imaging Fundamentals and Applications*; A K Peters: Wellesley, MA, USA, 2008.
25. McCamy, C.S.; Marcus, H.; Davidson, J.G. A colour rendition chart. *J. Appl. Photogr. Eng.* **1976**, *11*, 95–99.
26. Pascale, D. *RGB Coordinates of the Macbeth ColourChecker*; The BabelColour Company: Montreal, QC, Canada, 2006.
27. Sharma, G.; Wu, W.; Dalal, E.N. The CIEDE2000 Colour-difference formula: Implementation notes, supplementary test data and mathematical observations. *Colour Res. Appl.* **2005**, *30*, 21–30. [[CrossRef](#)]
28. Song, T.; Luo, M.R. Testing color-difference formulae on complex images using a CRT monitor. In Proceedings of the IS & T and SID Eighth Color Imaging Conference, Scottsdale, AZ, USA, 7–10 November 2000; pp. 44–48.
29. ISO 15739: 2003 Photography—Electronic Still-Picture Imaging—Noise Measurements. 2013. Available online: http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=59420 (accessed on 15 January 2016).
30. Motwani, M.; Gadiya, M.; Motwani, R.; Harris, F. Survey of image denoising techniques. In Proceedings of Global Signal Processing Expo Conference (GSPx), Santa Clara, CA, USA, 2004; pp. 27–30.
31. Patil, J.; Jadhav, S. A comparative study of image denoising techniques. *Int. J. Innov. Res. Sci. Eng. Technol.* **2013**, *2*, 787–794.
32. Lebrun, M.; Colom, M.; Buades, A.; Morel, J.M. Secrets of image denoising cuisine. *Acta Numer.* **2012**, *21*, 475–576. [[CrossRef](#)]
33. Nowak, R.; Baraniuk, R. Wavelet-domain filtering for photon imaging systems. *IEEE Trans. Image Process.* **1997**, *8*, 666–678. [[CrossRef](#)] [[PubMed](#)]

34. Kolaczyk, E. Wavelet shrinkage estimation of certain Poisson intensity signals using corrected thresholds. *Stat. Sin.* **1999**, *9*, 119–135.
35. Makitalo, M.; Foi, A. Optimal inversion of the Anscombe transformation in low-count Poisson image denoising. *IEEE Trans. Image Process.* **2011**, *20*, 99–109. [[CrossRef](#)] [[PubMed](#)]
36. Foi, A. Noise estimation and removal in MR imaging: The variance-stabilization approach. In Proceedings of the IEEE International Symposium on Biomedical Imaging: From Nano to Macro, Chicago, IL, USA, 30 March 2011–2 April 2011; pp. 1809–1814.
37. Imagenomic LLC. *Noiseware 5 Plug-In User's Guide*; Imagenomic LLC: Alexandria, VA, USA, 2012.
38. Petrosyan, A.; Ghazaryan, A. Method and System for Digital Image Enhancement. U.S. Patent 7751641 B2, 28 April 2005.
39. Seiz, G.; Baltsavias, E.P.; Grün, A. Cloud mapping from ground: Use of photogrammetric methods. *Photogram. Eng. Remote Sens.* **2002**, *68*, 941–951.
40. Buades, A.; Coll, B.; Morel, J.M. A review of image denoising algorithms, with a new one. *Multiscale Model. Simul.* **2005**, *4*, 490–530. [[CrossRef](#)]
41. Kervramm, C.; Boulanger, J. Optimal spatial adaptation for patch-based denoising. *IEEE Trans. Image Process.* **2006**, *15*, 2866–2878. [[CrossRef](#)]
42. Awate, S.P.; Whitaker, R.T. Unsupervised, information-theoretic, adaptive image filtering for image restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28*, 364–376. [[CrossRef](#)] [[PubMed](#)]
43. Lebrun, M.; Colom, M.; Morel, J.M. The noise clinic: A blind image denoising algorithm. *Ipol J.* **2015**, *5*, 1–54. [[CrossRef](#)]
44. Colom, M.; Buades, A.; Morel, J. Nonparametric noise estimation method for raw images. *J. Opt. Soc. Am. A* **2014**, *31*, 863–871. [[CrossRef](#)] [[PubMed](#)]
45. Lebrun, M.; Colom, M.; Morel, J.M. The noise clinic: A universal blind denoising algorithm. In Proceedings of the IEEE International Conference on Image Processing, Paris, France, 27–30 October 2014; pp. 2674–2678.
46. Lebrun, M.; Colom, M.; Morel, J.M. Multiscale image blind denoising. *IEEE Trans. Image Process.* **2015**, *24*, 3149–3161. [[CrossRef](#)] [[PubMed](#)]
47. Ponomarenko, N.N.; Lukin, V.V.; Zriakhov, M.S.; Kaarna, A.; Astola, J.T. An automatic approach to lossy compression of AVIRIS images. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, Barcelona, Spain, 23–28 July 2007; pp. 472–475.
48. Dabov, K.; Foi, A.; Katkovnik, V.; Egiazarian, K. Colour image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space. In Proceedings of the IEEE International Conference on Image Processing, San Antonio, TX, USA, 16 September–19 October 2007; pp. 313–316.
49. Dabov, K.; Foi, A.; Katkovnik, V.; Egiazarian, K. Image denoising by sparse 3D transform-domain collaborative Filtering. *IEEE Trans. Image Process.* **2007**, *16*, 2080–2095. [[CrossRef](#)] [[PubMed](#)]
50. Lowe, D. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [[CrossRef](#)]
51. Benedetti, L.; Corsini, M.; Cignoni, P.; Callieri, M.; Scopigno, R. Colour to gray conversions in the context of stereo matching algorithms: An analysis and comparison of current methods and an ad-hoc theoretically-motivated technique for image matching. *Mach. Vis. Appl.* **2012**, *23*, 327–348. [[CrossRef](#)]
52. Shizume, T.; Ohashi, G.; Takamatsu, H.; Shimodaira, Y. Estimation of the Helmholtz-Kohlrausch effect for natural images. *J. Soc. Inf. Disp.* **2014**, *22*, 588–596. [[CrossRef](#)]
53. Smith, K.; Landes, P.E.; Thollot, J.; Myszkowski, K. Apparent greyscale: A simple and fast conversion to perceptually accurate images and video. *Comput. Graph. Forum* **2008**, *27*, 193–200. [[CrossRef](#)]
54. Kim, Y.; Jang, C.; Demouth, J.; Lee, S. Robust colour-to-gray via nonlinear global mapping. *ACM Trans. Graph.* **2009**, *28*. [[CrossRef](#)]
55. Grundland, M.; Dodgson, N.A. Decolourize: Fast, contrast enhancing, colour to grayscale conversion. *Pattern Recognit.* **2007**, *40*, 2891–2896. [[CrossRef](#)]
56. Lu, C.; Xu, L.; Jia, J. Contrast preserving decolourization. In Proceedings of the IEEE International Conference on ICCP, Seattle, WA, USA, 28–29 April 2012; pp. 1–7.
57. Lu, C.; Xu, L.; Jia, J. Real-time contrast preserving decolourization. In Proceedings of the SIGGRAPH Asia 2012 Technical Briefs, Singapore, 28 November–1 December 2012; ACM: New York, NY, USA, 2012.
58. Lu, C.; Xu, L.; Jia, J. Contrast preserving decolourization with perception-based quality metrics. *Int. J. Comput. Vis.* **2014**, *110*, 222–239. [[CrossRef](#)]

59. Song, Y.; Bao, L.; Xu, X.; Yang, Q. Decolourization: Is rgb2gray() out? In Proceedings of the SIGGRAPH Asia 2013 Technical Briefs, Hong Kong, China, 19–22 November 2013; ACM: New York, NY, USA, 2013.
60. Ciocca, G.; Cusano, C.; Gasparini, F.; Schettini, R. Content-aware image enhancement. In Proceedings of the Artificial Intelligence and Human-Oriented Computing, Rome, Italy, 10–13 September 2007; pp. 686–697.
61. Kou, F.; Chen, W.; Li, Z.; Wen, C. Content adaptive image detail enhancement. *IEEE Signal Process. Lett.* **2015**, *22*, 211–215. [[CrossRef](#)]
62. Sarkar, A.; Fairchild, M.D.; Caviedes, J.; Subedar, M. A comparative study of colour and contrast enhancement for still images and consumer video applications. In Proceedings of the 16th Colour Imaging Conference: Colour Science and Engineering Systems, Technologies and Applications, Portland, OR, USA, 10–15 November 2008; pp. 170–175.
63. Baltavias, E.P. Multiphoto Geometrically Constrained Matching. Ph.D. Thesis, Institute of Geodesy and Photogrammetry, ETH, Zurich, Switzerland, December 1991.
64. Baltavias, E.P.; Li, H.; Mason, S.; Stefanidis, A.; Sinning, M. Comparison of two digital photogrammetric systems with emphasis on DTM generation: Case study glacier measurement. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **1996**, *31*, 104–109.
65. Ohdake, T.; Chikatsu, H. 3D modeling of high relief sculpture using image based integrated measurement system. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2005**, *36*, 6.
66. Remondino, F.; El-Hakim, S.; Gruen, A.; Zhang, L. Turning images into 3D models—Development and performance analysis of image matching for detailed surface reconstruction of heritage objects. *IEEE Signal Process. Mag.* **2008**, *25*, 55–65. [[CrossRef](#)]
67. MacDonald, L.; Hindmarch, J.; Robson, S.; Terras, M. Modelling the appearance of heritage metallic surfaces. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2014**, *XL-5*, 371–377. [[CrossRef](#)]
68. Jazayeri, I.; Fraser, C.S. Interest operators for feature-based matching in close range photogrammetry. *Photogramm. Rec.* **2010**, *25*, 24–41. [[CrossRef](#)]
69. Vedaldi, A.; Fulkerson, B. VLFeat—An open and portable library of computer vision algorithms. In Proceedings of the 18th ACM International Conference on Multimedia, Firenze, Italy, 25–29 October 2010.



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons by Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).