2016 Fourth International Conference on 3D Vision

Detecting and Correcting Shadows in Urban Point Clouds and Image Collections

M. Guislain*[1,2], J. Digne[1], R. Chaine[1], D. Kudelski[2], and P. Lefebvre-Albaret[2]

[1]LIRIS - CNRS UMR 5205, Université Lyon 1, France
[2]Technodigit, Genay, France

Abstract

LiDAR (Light Detection And Ranging) acquisition is a widespread method for measuring urban scenes, be it a small town neighborhood or an entire city. It is even more interesting when this acquisition is coupled with a collection of pictures registered with the data, permitting to recover the color information of the points. Yet, this added color can be perturbed by shadows that are very dependent on the sun direction and weather conditions during the acquisition. In this paper, we focus on the problem of automatically detecting and correcting the shadows from the LiDAR data by exploiting both the images and the point set laser reflectance. Building on the observation that shadow boundaries are characterized by both a significant color change and a stable laser reflectance, we propose to first detect shadow boundaries in the point set and then segment ground shadows using graph cuts in the image. Finally using a simplified illumination model we correct the shadows directly on the colored point sets. This joint exploitation of both the laser point set and the images renders our approach robust and efficient, avoiding user interaction.

1. Introduction

As more and more cities embrace the digital revolution, there is a growing demand for digitization technologies able to reconstruct urban scenes from geometric measurements and pictures. The goal might be, for example, to plan urban evolution or allow virtual tours of the city. A common way to perform these measurements is through a LiDAR acquisition campaign. Indeed, range laser scanners as well as cameras can be embedded aboard a vehicle moving in the streets, taking laser measures and pictures as it goes. The final result is a point cloud and a series of pictures that are either registered by calibration or by optimization algorithms. Thus the geometric information can be augmented with some color information. However, this information depends strongly on the lighting conditions during the acquisition which in turn depend on the daytime, season and weather. This leads to the question of retrieving illumination-free colors in order to have a time-independent urban scene acquisition. In urban scenes the most important problem lies in the shadows projected by the buildings or by the street furniture and trees.

In this paper we focus on shadow detection in the point cloud by jointly analyzing the geometry, reflective and color properties of the scene. The detection and correction of shadows is closely linked to the *intrinsic image* recovery problem, which decomposes an image into a reflectance and a shading layer [9]. However we will use a simplified illumination model compatible with our input data and avoiding user interaction. To do so, we build on a simple characterization of the shadow edge: two points lie on two sides of a shadow boundary if they have similar laser reflectances but very different color values. Indeed, the value of the laser reflectance depends solely on the reflective properties of the surface and not on the fact that the surface lies in the shadows or in an illuminated area.

Input data. We consider unstructured point clouds and associated pictures acquired from urban scenes. The point cloud is endowed with the laser reflectance intensity which depends on the reflective properties of the surface impacted by the laser. It is typically given by the amount of photon that flies back toward the LiDAR. The pictures GPS positioning is also available which will be important for the illumination correction step. Images are considered to be precisely registered relatively to the point cloud. They can be taken at the same time as the LiDAR acquisition process or at a different time, but they are assumed to be taken in a relatively small period of time so that all images have consistent illumination conditions.

To summarize, our contributions are the following:

- A shadow characterization that does not require user

*maximilien.guislain@liris.cnrs.fr

978-1-5090-5407-7/16 $31.00 © 2016 IEEE
DOI 10.1109/3DV.2016.63

interaction

- A new energy minimization formulation for segmenting shadows and sunlit areas
- The joint exploitation of the image structure and LiDAR data reflectance properties

The remainder of this paper is divided as follows: in section 2 we review relevant previous work. Section 3 gives an overview of our method. Then, sections 4 and 5 detail the methods for detecting and correcting the shadows. Finally, the results are shown in section 6.

2. Related work

Shadow detection and removal is already a widely explored field. Methods have been developed to detect shadows either on a single image, a sequence of images or datasets containing both geometric information such as a LiDAR point set and registered images. Each of these problem settings requires a specific problem formulation, although some common ideas can be found in all these approaches. We review below relevant work emphasizing concepts that are linked to our approach.

Single image shadow detection

The detection and correction of shadows in a single image is a difficult problem that has been extensively studied. We restrict our analysis to some recent works that not only propose methods to detect the shadows but also address the re-illumination problem.

Lalonde *et al*. [8] present a method to locate and remove the shadows based on machine learning. First the shadows are detected using a watershed segmentation and a Canny edge detection. Descriptors composed of histograms and skewness of the pixels intensities are analyzed on both sides of the detected edges. Shadow edges are then consolidated using a probabilistic model. Finally the scene layout (decomposition between ground, sky, and vertical surfaces) is added to only detect and consider the shadows laying on the ground.

Corke *et al*. [3] propose a method to recover a grayscale intrinsic image from a standard RGB image by considering the physical properties of the light and using also the *Log-Chromaticity* colorspace. In this particular colorspace, the colors of all materials under varying illuminations change along parallel directions. By projecting the points on a line perpendicular to these directions, an illumination-free color can be found. It however requires user interaction to define two regions of the same material under different illumination conditions and a perfect knowledge of the camera sensor response.

Xiao *et al*. [16] remove shadows from a single image using a multiscale illumination transfer in the La*b* colorspace. Shadowed and lighted regions are roughly sketched by the user and then segmented using a Gaussian Mixture Model. A multi-scale illumination transfer between the shadowed and lighted regions is then performed assuming lambertian surface illumination properties. Finally the shadow boundaries are reprocessed using a Bayesian framework to remove relighting artifacts.

Images sequence and 3D proxy

Over the past ten years, fast and robust photogrammetric methods have been developed to compute a 3D model from a series of pictures. Two of the most well known methods are Structure from Motion (SfM) [13] and Patch-based Multi-View Stereo (PMVS) [4]. These multiview methods have been exploited to enrich the information of 2D images in order to retrieve specific lighting properties.

Laffont *et al*. [7] propose a complete and precise process to retrieve the intrinsic characteristic from a set of images. This algorithm requires a set of HDR images, a set of LDR images from which a proxy 3D model is built, and a measured environment map. User interaction is also required to define the sun orientation and obtain two gray color values in the sunlight and in the shadow. This set of input data are used to decompose the illumination of each vertex of the proxy model into the original albedo and the sun, sky and indirect luminance. Finally these vertices are projected onto the image planes to retrieve the intrinsic images. Although this method proves efficient, it requires a lot of additional information that are not available in our context, in addition to user interaction, something we avoid.

In a somewhat similar manner, Wehrwein *et al*. [15] automatically detect the shadows and the sun direction in a series of picture. Applying SfM to the pictures yields a 3D model. A set of colors is associated to each vertex of the mesh that comes from all the images that see the vertex. From the statistics of the proposed colors, the algorithm can guess which points lie in the shadow in each image. These shadow point labels are used to consolidate the shadow edges and estimate the orientation of the sun.

Shadow detection on 3D LiDAR data

Detection of shadows on 3D LiDAR data is a less explored environment compared to single image and images collections.

Troccoli and Allen [14] relight a 3D point cloud model using multiple HDR overlapping images taken under different illumination conditions. Their strategy is to compute and analyze illumination ratio in overlapping areas and use these ratios to relight the whole image. This approach however requires a mesh reconstruction step to get rid of the point cloud and work on a watertight surface.

Ramakrishnan *et al*. [10] propose to correct the colors of a point cloud using a consistent and continuous illumination model by removing the direct sun illumination and normalizing the sky illumination. The indirect illumination from the other part of the scene is considered negligible. The sun orientation is obtained using the GPS coordinates of the scene as well as the known sun position relative to this position at the time of the scene acquisition. Similarly to [3], pairs of points of the same material are selected by the user both in sunlight and in the shadowed parts of the cloud. These pairs are used to compute the sun illumination and sky illumination contributions which yield an estimation of the illumination values for each point of the cloud. Unlike our approach, this method requires user interaction. Furthermore, this algorithm is tested on LiDAR scans measured on separated buildings, a setting quite different from our complex urban scene point clouds, where the buildings are concentrated and the streets narrow. Finally, this method relies heavily on having a clean 3D model. However in our context, the 3D model is only imperfectly measured, which can affect the quality of the sun visibility estimation (there can be missing building parts) and alter the final results.

3. Overview

Our approach takes as input a point cloud endowed with laser reflectance information as well as a series of pictures aligned with the point cloud. From this relatively common input data, we propose to detect the ground shadows and to correct them using a simplified illumination model. The main characteristic of our algorithm lies in the joint exploitation of both images and point clouds.

Our shadow detection algorithm proceeds in several steps:

- Shadow interfaces are identified using reflectance and luminance properties of the colored point cloud
- Interface points are filtered using local histogram analysis
- The shadows are segmented in the image plane using graph cut with well chosen graph edge costs

Once the shadows are detected the correction is a two-step method building on a simplified illumination model performing a luminance correction in the first step and a chrominance correction in a second step.

4. Shadow detection

Our shadow detection method relies on the observation that a shadow barely impacts the reflectance value of a point while it strongly impacts the color value. Indeed the laser reflectance only depends on the material properties of the surface whereas the color takes into account the illumination of a point. Therefore our first step is to look for shadow interface areas that have low reflectance gradients and high color gradients.

4.1. Shadow interface detection

In this first step, the algorithm works directly on the point cloud and finds pairs of close points that are likely to lie on each side of a shadow interface.

We define the neighborhood of a point as the set of its K nearest neighbors and estimate the color gradient ∇c and reflectance gradient ∇R in this neighborhood. Since the value of K impacts the computation time, we use a small value $K = 4$ at the risk of loosing some interfaces. As will be seen below, we still get enough interfaces to detect the shadows. The gradients between two neighboring points p and q are defined as follows:

$$\nabla R = \frac{|R_p - R_q|}{R_p + R_q}$$

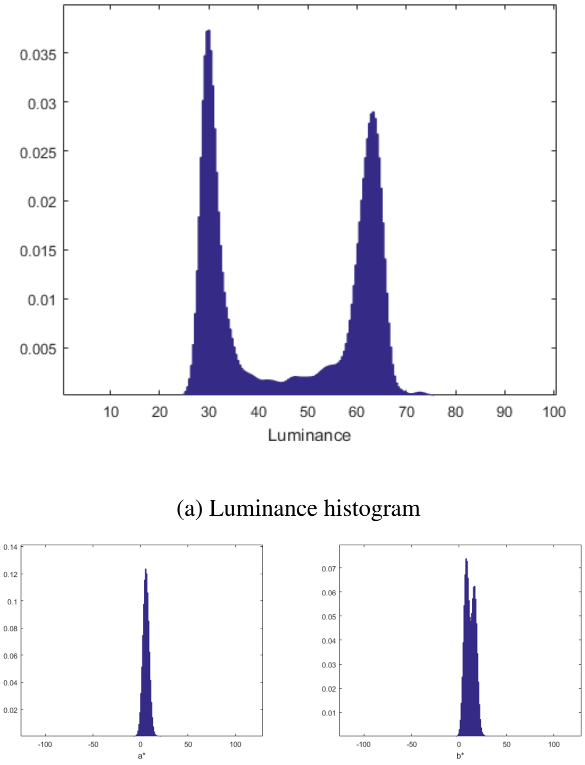
$$\nabla c = E_{00}(C_p, C_q)$$

where E_{00} is a weighted difference of the color expressed in the La*b* coordinates to render the gradient perceptually meaningful. The exact formula for E_{00} can be found in [12]. To detect relevant pairs of points we use a threshold that loosely selects a set of shadow interfaces. This threshold is set experimentally: two points are considered to lie at a shadow interface if $|\nabla c| \geq 2.33$ and $|\nabla R| \leq 0.05$. It may appear counterintuitive to use the color gradient instead of the luminance gradient since the luminance should contain all illumination information. However, in the case of outdoors scenes, not only the luminance but also the chromaticity of the points is affected by the sun visibility, as noted by Khan and Reinhard [6] and Corke [3]. Indeed, sunlit areas tend to be more yellowish (due to the sun light spectrum), while unlighted areas tend to appear more blueish (due to Rayleigh scattering). Therefore, it is better to use the full color difference rather than the luminance only. This assumption was verified experimentally by a higher number of meaningful detection.

Since our threshold is not tight, irrelevant detected pairs of points are filtered out based on a local density criterion: if a single pair is detected in a large area then it is likely that this pair is a false positive. In practice we remove the pairs of points that do not contain at least $1/4$ of their K' nearest neighbors as potential shadow interface ($K' = 21$). While this step does not remove all the outliers it is sufficient to reduce the computational burden of the next filtering step.

4.2. Shadow interfaces filtering

A further filtering of the shadow interfaces is achieved by analyzing the luminance histograms around a detected shadow interface pair of points. Indeed, histograms around a pair corresponding to a true shadow interface should have a bimodal distribution reflecting the light and shadow parts, as illustrated by figure 1.



(a) Luminance histogram

(b) a* histogram

(c) b* histogram

Figure 1: Histogram of the different La*b* color channels around a shadow interface.

Following this observation, points pairs should only be kept if their histograms indeed display a bimodal distribution. Luminance histograms are computed by considering points that lie close to the pair and exhibit similar normal and reflectance values. This avoids considering luminance coming from potentially different materials in the vicinity of the pair. Histograms are then smoothed using a convolution and the peaks are detected by computing the first derivative of the distribution function. Small peaks are discarded if their heights are less than 75% of the highest peak. The remaining number of peaks should be exactly equal to 2 for the distribution to be considered bimodal. In this case, the interface pair is kept. The values of the two peaks define L_L and L_S the luminance in sunlit areas and in the shadowed areas respectively.

4.3. Shadow consolidation

Previous steps allowed for the detection of interface points pairs which should next be turned into a complete segmentation method. To do so we momentarily set aside the point cloud, and exploit the detected interfaces directly in the image plane, since an image is a denser information. Thus, we project our shadow interface pairs in the image plane, using the camera known pose and intrinsic parameters.

Using these interface points to estimate the sun position and deduce the segmentation might look like an appealing idea but it would require a complete and consistent geometry to be able to retrieve the shadow mask. However if it is indeed the case on most terrestrial scans, some scanners may either miss occluding geometry or even do not account for all the geometry of the scene. For example, the Velodyn scanner of the KITTI dataset [5] measures points that are less than around 2 meters off the ground. We propose a different approach, relying on the images to compute a shadow segmentation: we look for a labelling δ of the image pixels, equal to 1 if the pixel lies in a shadow and 0 otherwise. The segmentation problem can be stated as an energy minimization with the following objective:

$$E(\delta) = \sum_{p\text{:image pixel}} E_{data}(\delta, p) + \gamma \sum_{(p,q)\text{: neighboring pixels}} E_{smooth}(\delta, p, q) \quad (1)$$

where $E_{smooth} = e^{-\frac{1}{2\sigma^2}(L(p)-L(q))}\mathbb{1}_{\delta(p)\neq\delta(q)}$,

$$E_{data}(\delta, p) = \begin{cases} e^{-\frac{1}{2\sigma^2}(L(p)-L_S)^2} & \text{if } \delta(p) = 0 \\ e^{-\frac{1}{2\sigma^2}(L(p)-L_L)^2} & \text{if } \delta(p) = 1 \end{cases}$$

and γ is a weighting term set to 1 in our experiments.

This type of energy can be easily minimized using graph cuts [2] [1] by using E_{data} as the source and sink edge costs and E_{smooth} as the inter-pixel edge cost. The labels obtained by the graph cut are then back-projected on the point cloud to assign the points to the shadow mask. On figure 2, we show the results of initial interface detection, interfaces filtered by density, by histogram analysis and the final graph cut result, demonstrating how the graph cut turns the sparse set of shadow edges into a full segmentation. This step retrieved the proper shadowed areas in a scene where the surface reflectance changes widely due to material variation (ie., white building wall vs darker building wall).

5. Shadow Correction

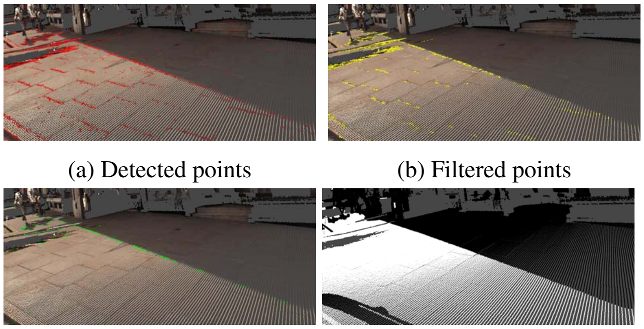
Once the shadows are segmented, the next step is to project the labelling on the point cloud, and to modify the color of the points so as to obtain a shadow-free point cloud.

5.1. Illumination model

According to [17], [7] and [10] the luminance of a point lying on an lambertian outdoor surface can be defined as:

$$L = R(S_{sun} + S_{sky} + S_{indirect}) \quad (2)$$

$$S_{sun} = V_{sun} cos\alpha E_{sun} \quad (3)$$



(a) Detected points (b) Filtered points

(c) Shadow interface points (d) Shadow mask

Figure 2: Points detected as potential shadow interfaces (2a), density filtered points (2b), shadow interface points (2c), and shadow mask from the graph cut. The shadow appearing at the bottom left of the graph cut mask (2d) is due to the acquisition vehicle that is visible on the pictures but not on the point cloud.

$$S_{sky} = \int_{\Omega_{sky}} cos\theta_{sky} E_{sky} \tag{4}$$

$$S_{indirect} = \int_{\Omega_{indirect}} cos\theta_{indirect} E_{indirect} \tag{5}$$

With I the illumination, R the albedo of the material, S_{sun} the lighting contribution of the sun, S_{sky} the lighting contribution of the sky (ambient light) and $S_{indirect}$ the lighting contribution of the light reflected by the objects in the scene. All notations are summarised on figure 3. Using this simplified model, our algorithm proceeds in two steps, correcting first the luminance component and second the chrominance component at each point.

5.2. Luminance correction

The luminance correction step requires an approximation of the sunlight orientation to be able to separate the different component of the scene lighting. Although the method proposed by [15] locates the sun orientation without *a priori* information, it cannot be applied on large urban scans due to the lack of precision on the surface to detect attached shadows. However, the sun azimuth and elevation can be estimated if the approximate GPS position and time of the data acquisition are known [11], leading to the sun orientation estimate. That is the method that was used to obtain the sun ray orientation relative to the scene. Knowing the position of several shadow points, the amount of illumination provided by the sky can be roughly estimated. Indeed, for shadow points the sun contribution can be safely ignored and the luminance of these points writes: $L = R(S_{Sky} + S_{Indirect})$, where R is the surface reflectance.

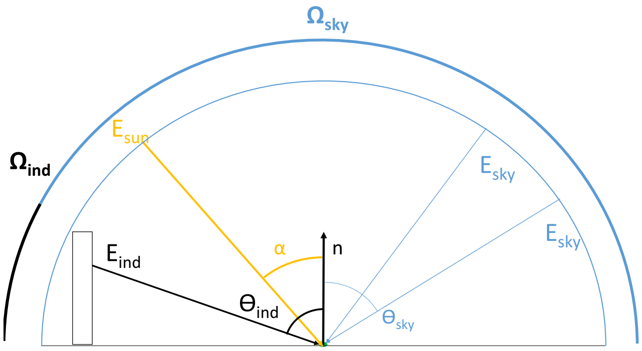


Figure 3: Illumination model of a point. The illumination is decomposed into 3 components: a sky contribution, a sun contribution and an indirect lightening contribution. Each sky ray ($\theta_{sky} \in \Omega_{sky}$) contributes an energy E_{sky}, the sun contributes an energy E_{sun} in direction α and the rays corresponding to indirect illumination ($\theta_{ind} \in \Omega_{ind}$) contribute an energy E_{ind}. In our simplified model we omit the indirect contribution.

We simplify the problem by considering that $S_{Sky} + S_{Indirect} = \beta \times E_{sky}$ where $\beta = 0.5 \times \langle \mathbf{n}(p), \mathbf{n}_{ground} \rangle + 1$. Thus β varies proportionally to the dot product of the ground normal $\mathbf{n}_{ground}$ and the point normal $\mathbf{n}(p)$.

This estimated lighting ratio β from the sky is error-prone in the case of urban environment where streets can be tightly enclosed by buildings, greatly reducing the real contribution of the sky compared to the contribution of indirect lightening. However it is still a good first approximation of the amount of light coming from the environment. It is thus possible to make the luminance uniform for shadowed points by computing $\bar{\beta}$ the average β in the shadowed area:

$$L'(p) = L(p)\frac{\bar{\beta}}{\beta(p)} \tag{6}$$

The matter is quite different for sunlit points. In a similar manner as [10] we define the sun-sky-ratio (SSR) of lighting between the sun and the sky:

$$SSR = \frac{\bar{L_L}\bar{\beta_S} - \bar{L_S}\bar{\beta_L}}{\bar{L_L}(\bar{cos\alpha})_L} \tag{7}$$

where α is the angle between the sun direction and the points normal. The straight bar above the notation means that the quantity is averaged on a light or a shadow area respectively, depending on the indice $_L$ or $_S$. More details on the derivation of this formula can be found in the original publication [10].

This ratio allowing us to recompute the luminance of points in sunlight:

$$L'(p) = L(p)\frac{\bar{\beta_S}}{SSR\cos\alpha + \beta(p)} \tag{8}$$

Applying these transformations to the luminance of the point successfully relight them in a realistic manner. Once the luminance is corrected, the chrominance can also be adapted.

5.3. Chrominance correction

The chrominance correction is simpler: it uses a simple chrominance ratio between the sunlit points and the shadowed points. It is performed for both the a and b components expressed in the La*b* colorspace.

$$b'(p) = b(p) \cdot \frac{\bar{b_S}}{\bar{b_L}} \text{ and } a'(p) = a(p) \cdot \frac{\bar{a_S}}{\bar{a_L}} \qquad (9)$$

This trivial update is enough for chrominance correction, as presented below in our experiments.

5.4. Penumbra zones

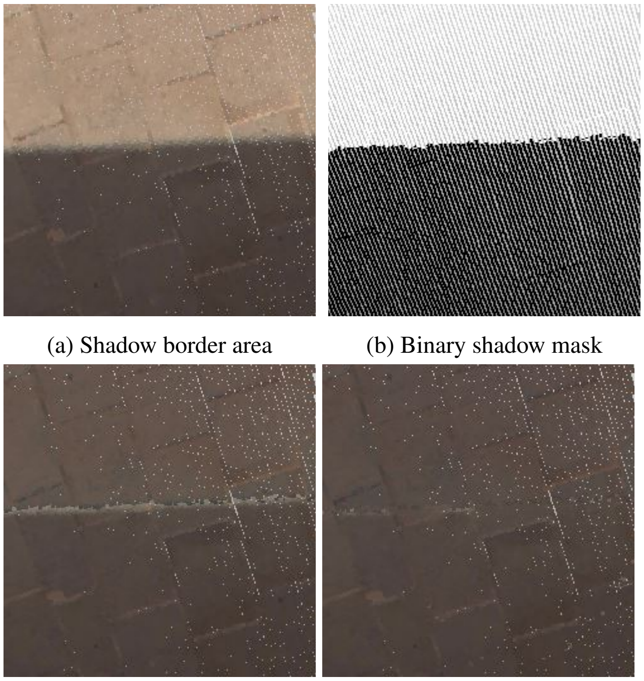
After this two-step correction, there may remain some artifacts near the boundaries of the shadowed and sunlit areas which should be corrected (figure 4). Indeed, having a binary shadow mask may lead to unwanted effects along the shadow edges where the boundary between light and darkness is not sharply defined. These effects are typically due to under- or over-compensation in the luminance and chrominance correction around the shadow edges, which induces the apparition of a strong linear artifact along the border. This artifact is clearly visible in figure 4c. To alleviate this effect we perform an in-painting step around the known shadow border. Since the area to inpaint is very small, we avoid any sophisticated inpainting method such as variational or patch-based inpainting and use a median filter to guess the missing pixel colors. This last step, despite not being perfect, mitigates the visual artifacts caused by overcompensation as depicted in figure 4d.

6. Results and Discussion

We demonstrate the efficiency of our shadow detection and correction schemes on two different sets of data. Both these sets are composed of points clouds with intensity values and registered images that are used to color the point cloud. The GPS positioning of the images is also available. In these two sets, images and point clouds have been acquired at the same time.

6.1. Leica Pegasus Dataset

Our main test-case is a dataset containing point clouds with reflectance values and associated images provided by Leica Geosystems, acquired in the city of Shrewsbury, UK. This point cloud offers smooth reflectances features that reflect well the luminance variation of the materials, almost without any noise. As can be seen in figure 5, it is possible to relight a point cloud by switching its dark points to a sunlight state or switching the sunlight points to a shadow state (see figure 5d).



(a) Shadow border area

(b) Binary shadow mask

(c) Re-lighted area.

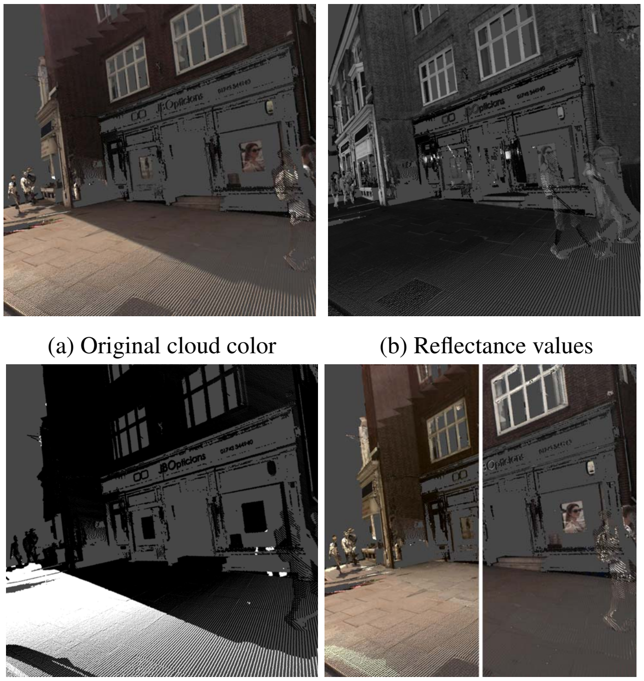
(d) Re-lighted area with local median filtering.

Figure 4: Small penumbra zones on the boundary of a shadow can have non negligible effect when re-lighting a point cloud.

Other locations displaying strong shadows were tested as well, as shown on figures 6, 7 and 8. In most cases the shadow is properly detected on the ground plane and extracted using the graph cut.

6.2. KITTI Dataset

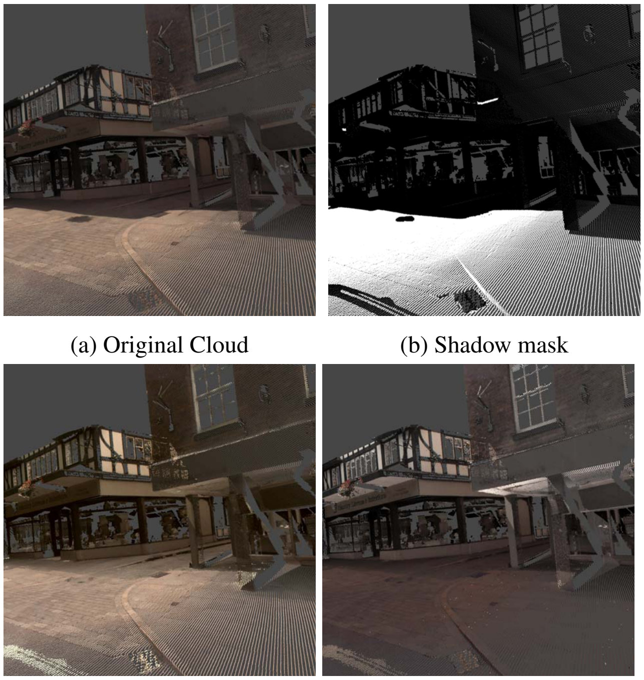
Our second test case is the KITTI Dataset [5]. Due to the nature of data, some pre-processing was performed to obtain a point cloud dense enough with smooth reflectance values. First a set of 20 scans around an image were merged together. From this merged point cloud, the point density was regularized. This point cloud was then colored using the color camera. This small pre-processing yielded a point cloud dense enough to be exploited jointly with the images. Our algorithm was then run with the standard parameters defined in sections 4 and 5. Figure 9 and 10 show that strong shadows were successfully detected by the process. However, some soft shadows located on very bright regions (*eg.*, the shadow of the tree on the white wall on the left) have not been correctly identified as shadows.



(a) Original cloud color (b) Reflectance values

(c) Shadow mask (d) Relighted cloud (light/dark)

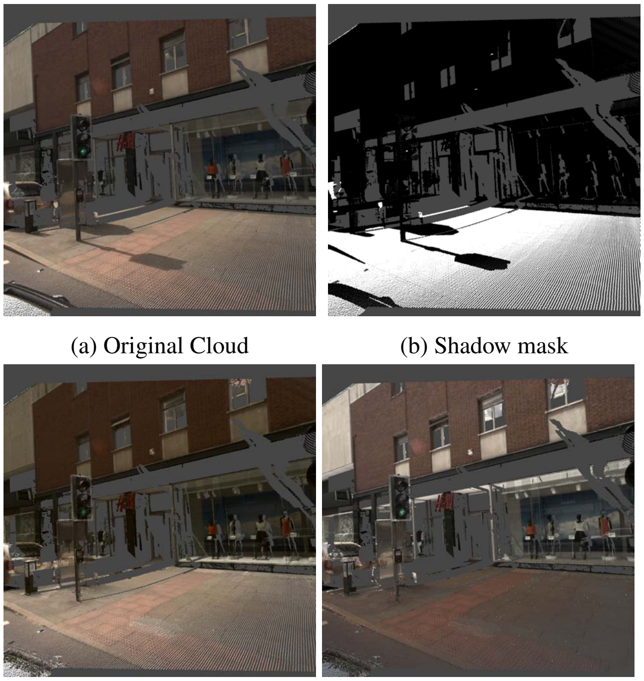
Figure 5: Comparison between the original color and the relighted colors of a fragment of the Pegasus point cloud where the shadows were suppressed.



(a) Original Cloud (b) Shadow mask

(c) Relighted cloud (light) (d) Relighted cloud (dark)

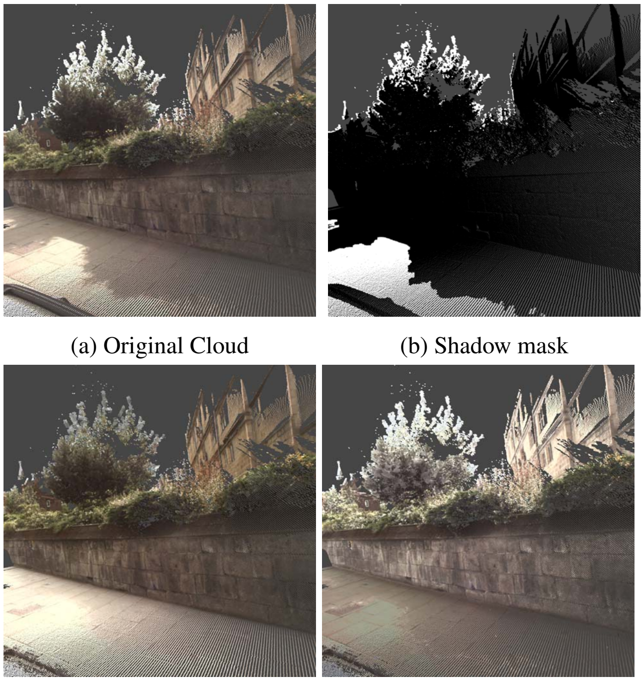
Figure 6: Comparison between the original color and the relighted colors of a fragment of the Pegasus point cloud



(a) Original Cloud (b) Shadow mask

(c) Relighted cloud (light) (d) Relighted cloud (dark)

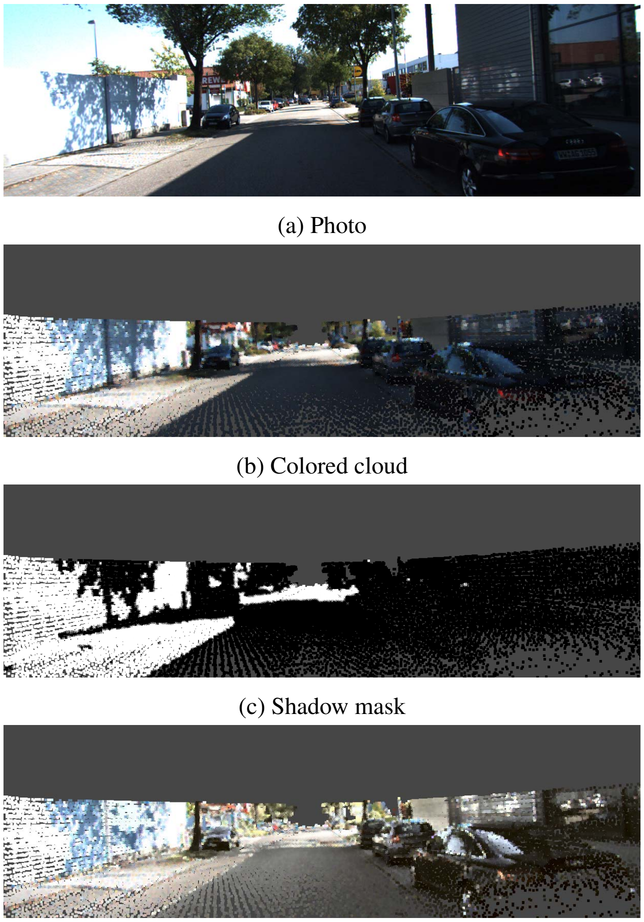
Figure 7: Comparison between the original color and the relighted colors of a fragment of the Pegasus point cloud



(a) Original Cloud (b) Shadow mask

(c) Relighted cloud (light) (d) Relighted cloud (dark)

Figure 8: Comparison between the original color and the relighted colors of a fragment of the Pegasus point cloud



(a) Photo

(b) Colored cloud

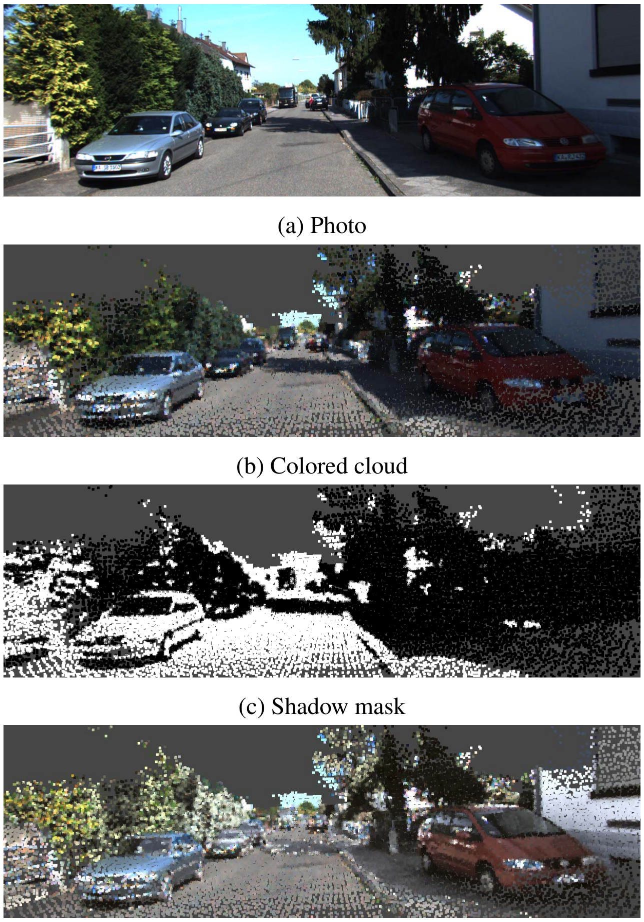
(c) Shadow mask

(d) Relighted cloud

Figure 9: Example of shadow detection and cloud relighting on the KITTI dataset

6.3. Discussion

As explained above, the sharp shadows can be correctly detected and corrected by our process. Unfortunately, in some cases, the relighted areas do not possess the exact same luminance and chromaticity as its original counterpart. This is slightly noticeable in figures 6 and 7 but is very clear on figure 9. Several factors may be responsible for this. First, the illumination model is only an approximation and does not perfectly reflect the illumination of the object (indirect and sky illumination). Second, the light source is supposedly uniform for each point of the cloud. However, illumination power may also vary in the case of a partially clouded sky. These two factors as well as the existence of soft shadows may account for the residual observable color difference. Another limitation lies in the lack of robustness to moving vehicles or pedestrians. They might not be at the same place in the image or in the point cloud which makes the joint exploitation fail.



(a) Photo

(b) Colored cloud

(c) Shadow mask

(d) Relighted cloud

Figure 10: Example of a shadow detection and cloud relighting on the KITTI dataset

7. Conclusion

We introduced a new shadow detection and correction algorithm that relies on the simple but efficient observation that reflectance is only marginally impacted by shadows whereas color information is. Our algorithm makes intensive use of both geometric information and color information working either on the point cloud or in the image plane in each step. Our simplified illumination model may lead however to some remaining artifacts, and in a future work we plan to explore more sophisticated models. Our method is not robust to moving objects, another research direction we want to further explore.

Acknowledgment

The authors acknowledge support from ANRT, PhD grant $n°2014/0319$.

References

[1] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, September 2004. 4

[2] Y. Boykov, O. Veksler, and R. Zabih. Efficient approximate energy minimization via graph cuts. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 20(12):1222–1239, November 2001. 4

[3] P. Corke, R. Paul, W. Churchill, and P. Newman. Dealing with shadows: Capturing intrinsic scene appearance for image-based outdoor localisation. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 2085–2092. IEEE, 2013. 2, 3

[4] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence*, 32(8):1362–76, Aug. 2010. 2

[5] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013. 4, 6

[6] E. A. Khan and E. Reinhard. Evaluation of color spaces for edge classification in outdoor scenes. In *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, volume 3, pages III–952. IEEE, 2005. 3

[7] P.-Y. Laffont, A. Bousseau, and G. Drettakis. Rich intrinsic image decomposition of outdoor scenes from multiple views. *IEEE transactions on visualization and computer graphics*, XX(X):1–16, Apr. 2012. 2, 4

[8] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan. Detecting ground shadows in outdoor consumer photographs. In *Computer Vision–ECCV 2010*, pages 322–335. Springer, 2010. 2

[9] E. H. Land and J. J. McCann. Lightness and retinex theory. *Journal of the Optical Society of America (1917-1983)*, 61:1, jan 1971. 1

[10] R. Ramakrishnan, J. Nieto, and S. Scheding. Shadow compensation for outdoor perception. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 4835–4842. IEEE, 2015. 2, 4, 5

[11] I. Reda and A. Andreas. Solar position algorithm for solar radiation applications. *Solar energy*, 76(5):577–589, 2004. 5

[12] G. Sharma, W. Wu, and E. N. Dalal. The ciede2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color Research & Application*, 30(1):21–30, 2005. 3

[13] N. Snavely, S. M. Seitz, and R. Szeliski. Modeling the world from internet photo collections. *International Journal of Computer Vision*, 80(2):189–210, 2008. 2

[14] A. Troccoli and P. K. Allen. Relighting acquired models of outdoor scenes. In *3-D Digital Imaging and Modeling, 2005. 3DIM 2005. Fifth International Conference on*, pages 245–252. IEEE, 2005. 2

[15] S. Wehrwein, K. Bala, and N. Snavely. Shadow detection and sun direction in photo collections. In *Proceedings of 3DV*, 2015. 2, 5

[16] C. Xiao, R. She, D. Xiao, and K.-L. Ma. Fast shadow removal using adaptive multi-scale illumination transfer. In *Computer Graphics Forum*, volume 32, pages 207–218. Wiley Online Library, 2013. 2

[17] Y. Yu, P. Debevec, J. Malik, and T. Hawkins. Inverse global illumination: Recovering reflectance models of real scenes from photographs. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 215–224. ACM Press/Addison-Wesley Publishing Co., 1999. 4