ACCURATE MULTI-VIEW DEPTH RECONSTRUCTION WITH OCCLUSIONS HANDLING

Cédric Niquin[12], *Stéphanie Prevost*[12], *Yannick Remion*[12]
{cedric.niquin|stephanie.prevost|yannick.remion}@univ-reims.fr

1 - CReSTIC SIC, Université de Reims Champagne-Ardenne, 51100 Reims, France
2 - TéléRelief, 8 rue Gabriel Voisin, 51100 Reims, France

ABSTRACT

We present an offline method for stereo matching using a large number of views. Our method is based on occlusions detection. It is composed of two steps, one global and one local. In the first step we formulate an energy function that handles data, occlusions, and smooth terms through a global graph-cuts optimization. In our second step we introduce a local cost that handles occlusions from the first step in order to refine the result. This cost takes advantage of both the multi-view aspect and the occlusions. The experimental results show how our algorithm joins the advantages of both global and local methods, and how much it is accurate on boundaries detection and on details.

***Index Terms*—** Stereo vision, Image sequence analysis, Three-dimensional displays, Minimization methods, Graph theory

1. INTRODUCTION

Stereoscopic is in vogue in many domains, as int the video games or the cinema with several movies diffused in three Dimension (3D). Autostereoscopic display is a new emergent technology that allows users to see in 3D without the use of glasses. This technology, more comfortable, could be used in many other domains like publicity for example. It also could be a way to bring 3D at home. This technology uses a number N of views, in opposition to the classical stereoscopic technology that only has two views. Nowadays and for most 3D autostereoscopic displays, N is equal to 8 or 9. One of the most impressive application with this kind of display is the augmented reality with depth reconstruction, which allow real objects to hide virtual ones. Another useful application is view synthesis, in order to generate all views for a display even from a small number of real views. Both of these applications are based on stereo-matching. This is the reason why multiview stereovision is a crucial field of research for the compagny TéléRelief, expert on technology of 3D displays. The most important aspect in this context is the multiview aspect, with a large number of views. We think that this aspect is a powerful tool for occlusions detection which is not well exploited in recent publications. This paper presents an offline method that computes the N depth maps in one single pass and works with any number of cameras ($N \geq 2$). It introduces a new method to fully integrate occlusions detection into the process of stereovision. It is composed of two steps: the first one is a global optimization step that performs occlusions detection in order to result in precise boundaries detection on the scene, and the second step is a local one that refines the depth maps.

2. RELATED WORK

Stereovision is divided into two distinct kinds of methods: local ones and global ones. The local methods do not give the best results but are non-expensive and can be implemented on GPU to run in real-time, whereas the global methods, based on energy minimization, give better results in an offline context. Our method, which combines the advantages of both kinds of methods, belongs to the global ones.

Several methods presented in recent papers use a segmentation on colors. This is the case of Wang and Zheng [1] which use the Mean-shift algorithm to segment their reference image into regions. However all photographies may not be adapted to segmentation, particularly the ones with many textures or not enough color variations. Since we want our method to be generic and to give good results in all contexts, we chose not to use segmentation in our algorithm. Figures 4(a) and 5(a) are examples of photographies we use.

The aim of global methods is to find the function f which minimizes an energy function $E(f)$, where f is a correspondence function which associates any pixel p from an image to a pixel f_p on another image. $E(f)$ is an energy function that associates a cost to any function f. There are several methods to approximate the f function minimizing $E(f)$. We use the graph-cuts method which is a well-known method to solve in a relatively fast way functions of the form of the function f that we introduce in the next section. Boykov and al. [2] present Graph-cuts and introduce two algorithms called "expansion move" and "swap move". The aim of these algorithms is to divide the energy minimization problem into several binary problems in order to permit the construction of a graph to solve it. Each node of the graph is a pixel and links between nodes have costs that describe the energy function. The energy can be minimized on a graph using the min-cut/max-flow algorithms. Most of these algorithms in the context of image analysis are presented and compared by Boykov and Kolmogorov [3]. Functions f that can be minimized are presented by Kolmogorov and Zabih [4].

Most of global methods use an energy function composed of two terms: the data term E_{data} (also called error term) and the smooth term E_{smooth}. The energy associated to a function f is then described as

$$E(f) = E_{data}(f) + E_{smooth}(f). \tag{1}$$

This kind of energy function is also used in scanline optimization [5]. The smooth term makes sure that the function f is smooth everywhere in global methods, and only horizontally in scanline optimization. The aim of the data term is to measure how appropriate the correspondence function f is for the images. In the case of two images, it is typically defined as

The work reported in this paper was supported in part by the French National Agency of Research as part of the CamRelief project. This project is a collaboration between the University of Reims Champagne-Ardenne and TéléRelief. We wish to thank Didier Debons, Michel Frichet and Florence Debons for their contribution to the project.

978-1-4244-4318-5/09/$25.00 ©2009 IEEE

$$E_{data}(f) = \sum_{p \in P} D(p, f_p), \tag{2}$$

where P is the set of all the pixels of the images and f_p is the pixel associated to p according to the function f. $D(p, f_p)$ generally is the squared difference or the absolute difference between the intensity of p and f_p. In the case of a number of photographies larger than 2, this energy is typically

$$E_{data}(f) = \sum_{p \in P} \sum_{i=1}^{N-1} D(f_p^i, f_p^{i+1}), \tag{3}$$

where N is the number of images and f_p^i is the pixel of image i associated to p according to f. The problem of this energy is that it does not take occlusions into account. The effect of occlusions is that a pixel associated to f_p^i into image i is not necessarily associated to f_p^{i+1} into image $i + 1$.

In order to reduce the effect of occlusions, Woetzel and Koch [6] introduce two local energies based on selection of a number M of image couples $(i, i + 1)$ into the $N - 1$ possible couples. The first possible selection is composed of the M minimum costs D. The second is made of the images either on the most left ($1 \leq i \leq M$), or on the right ($N - 1 - M \leq i \leq N - 1$). The means to select the M costs and the value of M are not well justified. Zhang and al. [7] directly integrate occlusions handling into their data term using a geometry constraint. However their context is completely different from ours since they use free moving cameras. It result to a completly different occlusions detection method than the one we define. Kolmogorov and Zabih [8] integrate the occlusions detection by means of the addition of a penality term to $E(f)$. They improve the resulting depth map, being more precise on boundaries. The data term is still defined as in (3) and thus, detected occlusions are not entirely taken into account. We present in section 3.1 a term E_{line} that merges occlusions detection, and a data term which fully integrates occlusions into data penality. Then in section 3.2 we introduce a selection of costs taking occlusions into account in a second step.

3. OUR METHOD

Our method is composed of two steps. The first is based on global optimization. We introduce a new energy function designed to detect occlusions in the scene, and its corresponding graph. The originality of the graph that we construct is that it connects pixels from different images, in order to optimize occlusions detection. To solve the energy minimization, we use graph-cuts with the "expansion move" algorithm as described by Boykov and al. [2]. The min-cut/max-flow algorithm that we use is the one given by Boykov and Kolmogorov [3], which seems to be faster in visualization problems. Then we present our second step, the refinement, which is a local method that takes advantage of results from the first step. To do that, we introduce a new cost taking occlusions into account. We assume that epipolar lines are paralleles and horizontal (either by capture system or by preprocessing) in order to express our algorithm as an horizontal disparity search. Disparity is the difference between the coordinates of two correspondant pixels along the scanline. Depth values are obtained by a triangulation step on disparities, using camera features.

3.1. Global Optimization

In this section, we present a new energy function to be minimized based on graph cuts. It is of the form

$$E(f) = E_{line}(f) + E_{v-smooth}(f), \tag{4}$$

where $E_{line}(f)$, combination of the local costs and occlusions, aims at making sure of the scanline optimization. The term $E_{v-smooth}(f)$ is used to add a constraint on vertical smoothing of f.

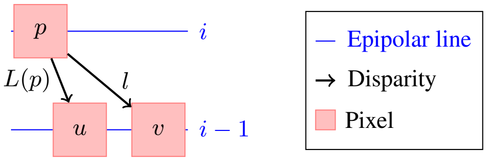


Fig. 1. Two possible correspondents for pixel p from image i.

More precisely, $E_{line}(f)$ is composed of two costs, summed on all the pixels

$$E_{line}(f) = \sum_{p \in P} \left(C(p, f_p^{I_p - 1}) + C(p, f_p^{I_p + 1}) \right), \tag{5}$$

where I_p is the image which contains pixel p and C is a cost which compare a pixel p with its corresponding pixels on the previous image or on the next one, according to f. Its value is either the dissimilarity penality is both pixels have the same disparity, or the occlusion penality if they do not. It is defined as

$$C(p, q) = \delta_{f_q^{I_p}, p} \times D(p, q) + (1 - \delta_{f_q^{I_p}, p}) \times C_{occ}, \tag{6}$$

where $\delta_{x,y}$ is a Kronecker's delta equal to 1 if x is equal to y, and 0 otherwise. C_{occ} is a constant real corresponding to the occlusion penality. We know that q is equal to either $f_p^{I_p - 1}$ or $f_p^{I_p + 1}$. So if the corresponding pixel of q is p, that means they both have opposite disparities. According to the Left/Right Checking (LRC) condition presented by Egnal and Wildes [9], we can deduce that there is no detected occlusion and $C(p, q)$ is then equal to $D(p, q)$, which could be the square difference or the absolute difference of their intensities. If p and q do not have exactly opposite disparities, the LRC condition failed, there is an occlusion and $C(p, q)$ is then equal to the constant C_{occ}. The value C_{occ} has to respect the following constraint to make sure that a graph describing this cost is constructible

$$C_{occ} \geq D(p, q). \tag{7}$$

Let's see how to construct the new graph corresponding to the cost $C(p, q)$, linking pixels from an image i to pixels from another one, either $i + 1$ or $i - 1$. The "expansion move" algorithm we use is composed of as many steps as the number of disparities that we want to assign to pixels. At each step all pixels p in P already have a disparity $L(p)$. The goal of a step is to check, for a disparity l, if the pixels have to keep their disparities or to change to disparity l in order to reduce the energy. Let's consider a graph composed of two terminals, called Source s and Sink t, and one node $n(p)$ for each pixel p of P. A cut in the graph separating a node from the source means that the pixel corresponding to this node will keep the same disparity. However a cut between the node and the sink means that the pixel will change its label to the current disparity l. In this context, illustrated by figure 1, a pixel p from image i could have two distinct correspondents on the image $i - 1$. Either p keeps the same disparity and its correspondent is a pixel u, or it changes to disparity l and then p becomes connected to v. When $L(p)$ is not equal to l , which means u and v are different, we have thus to add two connections between $(n(p), n(u))$ and $(n(p), n(v))$ to the graph.

First, let's consider the connection $(n(p), n(u))$. Since we have two terminals and two nodes to connect, there are four possible cuts to integrate into the graph, as figure 2 shows: $n(p)$ and $n(u)$ are both cut from s (C_{ss}^{pu}), $n(p)$ is cut from s and $n(u)$ from t (C_{st}^{pu}),

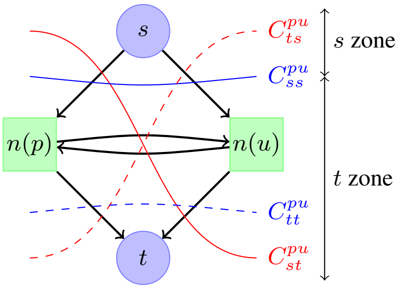


Fig. 2. Four possible cuts in a connection $(n(p), n(u))$ with example of s and t zones for the C_{ss}^{pu} cut.

then C_{ts}^{pu} and C_{tt}^{pu} which are defined in the same way. The cost of a cut is equal to the sum of all costs of the links (black arrow) which are crossed and go in the direction from the s zone to the t zone (see figure 2). We already know that the connection between p and u only exists if p keeps the same label, which means that any cuts going through the link between $n(p)$ and t have a null cost. Thus we have C_{ts}^{pu} and C_{tt}^{pu} equal to 0. For the cost C_{ss}^{pu}, both nodes keep the same disparity, this cost is equal to $D(p,u)$ if $L(p)$ is equal to $L(u)$, and C_{occ} otherwise. In the same manner, C_{st}^{pu} is equal to C_{occ}. We can then write C_{ss}^{pu} is equal to A and C_{st}^{pu} is the sum of A and B with

$$A = \delta_{L(p),L(u)} \times D(p,u) + (1-\delta_{L(p),L(u)}) \times C_{occ},$$
$$B = \delta_{L(p),L(u)} \times (C_{occ} - D(p,u)) .$$

A has to be applied to both C_{ss}^{pu} and C_{st}^{pu}. The only link that these costs have in common is the link between s and $n(p)$ (see figure 2). Thus the cost of this link is equal to A. B has to be applied only to the cut C_{st}^{pu}, and the link going from $n(u)$ to $n(p)$ is the only one that is not crossed by any other cut. B is then the cost of this link, as shown in figure 3(a).

Now let's consider the connection $(n(p), n(v))$. C_{ss}^{pv} and C_{st}^{pv} are null since p has to be associated to disparity l. The cost C_{tt}^{pv} is always equal to $D(p,v)$. C_{ts}^{pv} is equal to $D(n(p), n(v))$ if $L(v)$ is equal to l or C_{occ} otherwise. We have C_{ts}^{pv} equal to the sum of C and D, and C_{tt}^{pv} equal to C with

$$C = D(p,v),$$
$$D = \delta_{L(v),l} \times (C_{occ} - D(p,v)) .$$

C has to be applied for all cuts separating $n(p)$ from the sink, and D has to be only applied to the cut C_{ts}^{pv}. Figure 3(a) shows the final weighted graph with nodes $n(p)$, $n(u)$, $n(v)$ and costs of the links, when l is not equal to $L(p)$.

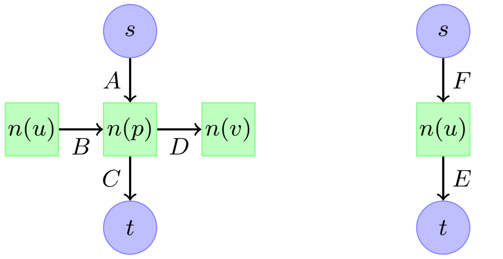
When l is equal to $L(p)$, u and v are the same pixel. Then the cost only depends on the disparity associated to that pixel, which we will call u in the following. The cost C_t^u is equal to E and C_s^u is equal to F with

$$E = D(p,u),$$
$$F = \delta_{L(u),l} \times D(p,u) + (1-\delta_{L(u),l}) \times C_{occ}.$$

Figure 3(b) shows the graph in this case.

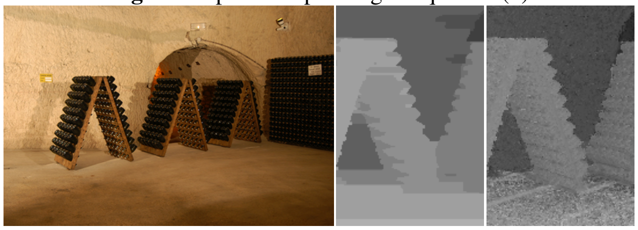
We have seen the first term of equation (4), $E_{line}(f)$, which integrates the penality to occlusions that occurs when one changes disparities. That means this term also makes sure that the function f is smooth on epipolar lines. We introduce a second term $E_{v-smooth}(f)$ to add a constraint on vertical smoothing of f. It is of the form

$$E_{v-smooth}(f) = \sum_{\{p,q\}\in N_{ei}} V\{p,q\}(f_p, f_q), \tag{8}$$



(a) $L(p) \neq l$ (b) $L(p) = l$

Fig. 3. Graph corresponding to equation (6).



(a) Champagne Ruinart. (b) Graph-cuts. (c) Refinement.

Fig. 4. Results obtained with cellar photographies.

where V is the absolute difference of the disparities associated to p and q, according to f. N_{ei} is the set of pairs of vertically adjacent pixels. Kolmogorov and Zabih [4] explain the way to construct the corresponding graph of $E_{v-smooth}(f)$.

At the end of the first step we finally have a correspondence function f, illustrated in figure 4(b), which shows precise boundaries of the scene but has undesirable stair step artifacts.

3.2. Local Refinement

In order to refine the results of our first step, we add a second step which is a local one, taking advantage of the precision of local methods. The main problem of these methods is to find the local cost which best describes the stereovision problem. Woetzel and Koch [6] introduce a cost based on the selection of a given number of image couples. We use a similar cost, but the originality of our method is that we use the results of the previous step to optimize the selection of the image couples. Let's define a correspondence function g as the result of the refinement step. The local cost $C_{local}(g)$ of function g on a pixel p is of the form

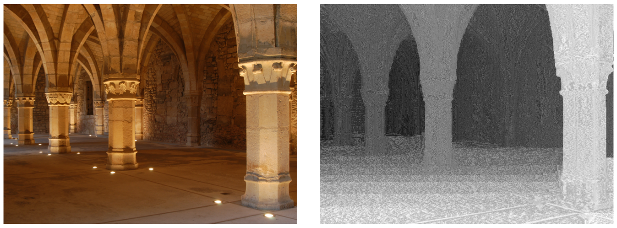
$$C_{local}(g) = \frac{1}{M} \times \sum_{i=1}^{N-1} \left(D(g_p^i, g_p^{i+1}) \times \delta_{f^i_{f_p^{i+1}}, p} \right), \tag{9}$$

where M is the number of image couples $(i, i+1)$ that verify the LRC condition $(f^i_{f_p^{i+1}} = p)$.

For each pixel, we compute the local cost of a finite number of disparities. We choose the disparities that verify $|f_p - g_p| \leq K$, where K is the maximum distance in pixels between two correspondents from f and g. The value of K is chosen by the user (see section 4 for a discussion about examples and consequences of different values of K). We use the Winner-take-all algorithm to select the best disparities: for each pixel p in P we select the disparity which minimizes $C_{local}(g)$. This step permits to refine the results from the previous one and to remove the stair step artifacts, as illustrated in figure 4(c).

4. RESULTS

In this section, we will discuss of the results we have and the effect of the different constant values of our algorithm. Figure 4(a) illus-



(a) Palais du Tau. (b) K=1.5, 30 disparities.

Fig. 5. Examples of results with Palais du Tau photographies.

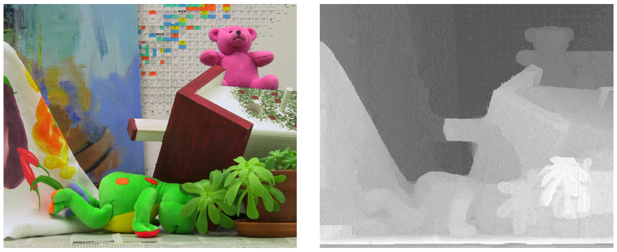
trates a series of 8 photographies taken at the cellar of champagne *Ruinart*. Images resolutions are 512×340. Figures 4(b) and 4(c) are extracts from the results we got after the two steps. For both steps, 30 disparities have been tested, with a value of K equal to 1.5. Figure 5(a) shows another series of 8 photographies, of resolutions 640×480 pixels, taken at *Palais du Tau* in Reims. Figure 5(b) is the result we obtain after the second step, with 30 disparities tested for both steps and a value of K equal to 1.5. Our method takes about 6 minutes to compute the 8 disparity maps of this last series, using a computer with 2.6 GHz and 2 Go of RAM. The second step is implemented using CUDA on an NVIDIA Quadro FX 3700.

The value of C_{occ} is a crucial choice, since its goal is to allow the detection of occlusions. If its value is too small, the resulting function f will not be continue enough on the horizontal direction, and occlusion will not be well detected. If the value is too high, changing depth will cost too much and the stair step artifacts will appear more and more. Empirically, we found that the value of 150 is a good compromise that gives good results on many of our tests ($D(p, q)$ is clamped to the value of C_{occ} in order to ensure equation 7). Whatever the choice of the value of C_{occ}, resulting depth maps will always look like a succession of vertical planes, because of graph-cuts. The value of K, in the second step, allows to correct these artefacts. The larger the value of K is, the more the stair step artifacts disappear, but the more noise appears in the depth maps.

Results on Teddy's images taken from Middlebury website [1] are shown in figures 6(a) and 6(b), with K equal to 0.8. We can see in figure 6(b) a default of our method, on the right side of the Teddy bear. This part of the image does not have any texture, thus, as a consequence of this lake of information, our first step assigns the same disparity than the one assigned to the bear, in order to avoid the addition of an occlusion (which is non-free). This kind of problem could be solved using a color segmentation on the images which allows to consider the bear and the wall behind it as two distinct regions. These methods give better results than our method in this context. However they are strongly dependent of the presence of a lot of distinct colors in the images. Indeed our own photographies, *Champagne Ruinart* and *Palais du Tau*, do not have many color variations, and do not give good color segmentation results. It is usually the same case, when shooting outside for example. This is the reason why we do not perform any color segmentation in our algorithm in order that our method may not be dependent of color variations, and still remain effective on our photographies, as shown in figures 4 and 5.

5. CONCLUSION AND FUTURE WORK

We have presented a new offline algorithm to compute N depth maps in one single pass, taking advantage of both global and local methods. It gives better contribution on multiview context, with a large number of views, even if it is still working in classical stereovision, with two views. Our algorithm is centered on occlusions detection, which is the primary problem of stereovision, using the Left-Right Checking condition. To do that we have introduced a new energy fonction and its corresponding graph, that has the particularity to link pixels from different images, to be minimized using graph-cuts. We also have introduced in our second step a local cost that handles results of our global method. Thanks to our refinement step, the result we got is accurate in objects boundaries as well as in details. Yet, our results are a bit noisy, and currently we search for a way to integrate aggregation to our results without affecting precision on occlusions. Another idea is to add weigths to image couples. Then our results will be exploitable in various domains like augmented reality or view synthesis. This last example is an important field of research where we think occlusions detection is also crucial. Knowing precisely the positions of occlusions is important in order to decide if we have to peek information from the left image or the right one. This is the reason why we will also work on the exploitation of the results from both steps of our algorithm in a view synthesis application.



(a) Teddy. (b) K=0.8, 16 disparities.

Fig. 6. Examples of results on images of Middlebury website.

6. REFERENCES

[1] Zeng-Fu Wang and Zhi-Gang Zheng, "A region based stereo matching algorithm using cooperative optimization," *IEEE Conference onComputer Vision and Pattern Recognition*, pp. 1–8, 2008.

[2] Yuri Boykov, Olga Veksler, and Ramin Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 1222–1239, 2001.

[3] Yuri Boykov and Vladimir Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 1124–1137, 2004.

[4] Vladimir Kolmogorov and Ramin Zabih, "What energy functions can be minimized via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 147–159, 2004.

[5] Christopher Zach, Mario Sormann, and Konrad Karner, "Scanline Optimization for Stereo on Graphics Hardware," in *Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission*, Washington, DC, USA, 2006, pp. 512–518, IEEE Computer Society.

[6] Jan Woetzel and Reinhard Koch, "Real-time multi-stereo depth estimation on GPU with approximative discontinuity handling," *1st European Conference on Visual Media Production*, pp. 245–254, 2004.

[7] Guofeng Zhang, Jiaya Jia, Tien-Tsin Wong, and Hujun Bao, "Recovering consistent video depth maps via bundle optimization," in *CVPR*. 2008, IEEE Computer Society.

[8] Vladimir Kolmogorov and Ramin Zabih, "Multi-camera scene reconstruction via graph cuts," in *European Conference on Computer Vision*, 2002, vol. 3, pp. 82–96.

[9] Geoffrey Egnal and Richard P. Wildes, "Detecting Binocular Half-Occlusions: Empirical Comparisons of Five Approaches," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1127–1133, 2002.

[1] http://vision.middlebury.edu/stereo