A FAST ALGORITHM FOR OCCLUSION DETECTION AND REMOVAL

Xiaoyi Yang, Yann Gousseau, Henri Maître, Yohann Tendero

LTCI, Telecom ParisTech, Université Paris-Saclay, 75013, Paris, France

ABSTRACT

This paper describes a simple and fast algorithm for removing occlusions that may occur in multiple views of a scene. In contrast to many methods of the literature, no assumption is made on occlusion shapes, colors or motions. Instead, this new method assumes that the background can be re-warped using an homography and that the reflectivity is quasi-Lambertian. After geometric and photometric alignments, three methods are evaluated. A median based method, a novel algorithm based on maximal clique detection and a robust PCA method are compared on real and simulated image sequences. This comparison shows that the new clique-based method provides the best performance in terms of quality and reliability.

***Index Terms*—** Image reconstruction, multi-image processing, mask removal, occlusion detection, background estimation, non-linear filtering.

1. INTRODUCTION

When photographing a famous monument or a scenery, many people experience the difficulty that some object occludes the scene or that someone wander into the view they wish to capture. Often, by the time one person moves out another one moves in. In such a situation, taking a picture without occlusions becomes tricky and time-consuming. This paper, as illustrated in figure 1, aims at proposing a simple, fast and reliable method to solve this problem by combining several photographs.

Several works have yet addressed this problem. Some authors focus on specific kinds of obstacles such as raindrops [1, 2, 3], reflections [4, 5, 6, 7], grids and fences [8, 9]. Some authors use a very dense sequence of images to deduce a depth map by optical flow and keep the farthest away surface [10]. Alternatively, some authors rely on specific dense sensor configurations that allow for a statistical decision [11]. If areas of the scene are never observed, an ultimate strategy consists in inpainting missing areas [12, 13] after a mask detection scheme has been performed. Yet, inpainting strategies are prone to errors, yielding artefacts in the reconstructed background. In addition, in many situations the framerate is not high enough to allow for a reliable optical flow computation or the assumptions on the mask shapes don't hold true. For all of theses reasons, we believe that a simple, fast and reliable algorithm should be proposed.

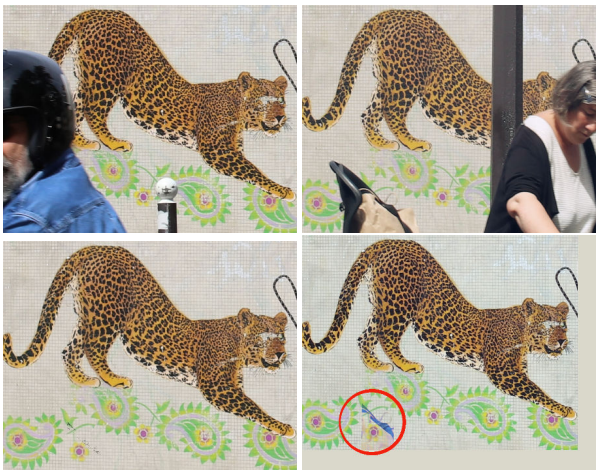

Fig. 1. On top: two frames of an image sequence after geometric and photometric alignments. Bottom left (resp. right) the proposed solution, (resp. right) the RPCA method. The red circle enlights a defect in the RPCA reconstruction.

Roughly, the solution proposed in this paper relies on the combined motions of the photographer and the masks to ensure that the sequence reveals the entire background one or several times. By geometrically and photometrically aligning the images we form a stack of images. Consequently, for each pixel we obtain a stack of values and decide what the background is. As we shall see, the method proposed in this paper assumes no specific shape, color, motion or texture for the occlusions. In addition, the proposed algorithm is simple, fast and, as we shall see in section 4, compares advantageously with a more sophisticated approach such as robust PCA [14].

Outline of the paper: Section 2 details the geometric and photometric alignments of image sequences. Section 3 discusses a new method for background detection based on a meaningful clique detection. Section 4 provides a comparative evaluation of the proposed algorithm and of two related methods: a median based method and a robust PCA (RPCA) based method. Section 5 discusses further refinements. *A webpage with an implementation, image sequences and numerical results is available [15].*

2. PROPOSED METHOD

We first detail the assumptions we shall use in section 2.1. Subsections 2.2 and 2.3 detail the methods used to align geometrically and radiometrically the image sequences.

2.1. Framework and Assumptions

As we have seen in section 1, many methods in the literature consider specific kinds of obstacles or occlusions such as raindrops, fences or grids. In contrast, we posit assumptions on the object of interest that we shall call hereinafter underlying background or just background. We suppose that we are given an image sequence such that the background 1) is quasi-Lambertian and 2) can be re-warped to a given reference image. Assumption 2) holds true if the scene is planar, like in our experiments, or if the camera undergoes a rotation around its optical center. Note that we do not make any assumptions on the background content, its color distribution, continuity or texture. Instead, we expect it to be quasi-Lambertian so that there is no significant color difference when looking from different positions. A limitation of the above assumptions is that the algorithm proposed in this paper cannot be expected to perform well when observing background with reflecting surfaces or specular reflectors like mirrors. We expect almost constant lighting conditions during acquisition.

2.2. Geometric Alignment

The approach we employ for this step is straightforward. A reference frame is chosen. The assumption 2) allows us to resample the observed frames on the reference frame using an homography [16] and bi-linear interpolation. The homography parameters are computed using RANSAC [17] on SIFT matches [18][1]. We expect that the homography with the largest number of matches corresponds to the background.

2.3. Chromatic Alignment

Under the quasi-Lambertian assumption, we expect the different images to have close colorimetric values at seen background pixels. However, we experience differences that depend on many uncontrolled factors between images. In addition, the camera white-balance algorithm also tends to modify the color content between images. The observed color distribution depends not only on the background but also on the masks or occlusions. Thus, we cannot use standard color transfer algorithms [19] to equalize the images. To solve the problem, we determine color transfer mappings between images. Indeed, digital camera conversion from input intensities to output vectors can be approximated by an invertible function [20]. Following [21], we use an order two polynomial model to compute this color transfer mapping. As we shall see, this choice often gives good results. Experiments show a poor correlation between channels. Hence, we compute a mapping for each color channel independently (in agreement with standard white-balance algorithms). The polynomial coefficients are computed from three pairs of SIFT matches, obtained from the geometric alignment, using a RANSAC strategy to robustify the selection.

[1] A SIFT match is defined by distance to the 1st neighbor $\leqslant$ 15 $\times$ distance to the 2nd neighbor.

3. IMAGE RECONSTRUCTION

After the geometric and photometric alignments described in sections 2.2 and 2.3, we obtain a stack of aligned images

$$\Phi(\mathbf{x}) = \{\mathbf{I}_i(\mathbf{x}),\ i \in \{1, \dots, n\}\} \tag{1}$$

defined $\forall \mathbf{x} \in \Omega \subset \mathbb{R}^2$, where $\forall \mathbf{x}$, $\mathbf{I}_i(\mathbf{x}) \in \mathbb{R}^3$. To estimate the background, for each pixel $\mathbf{x} \in \Omega$, we need to decide which value $\tilde{\mathbf{I}}(\mathbf{x})$ represents best the background. In this section, we detail two possible strategies to estimate $\tilde{\mathbf{I}}(\mathbf{x})$. The first one uses a median-based decision criterion (section 3.1). Section 3.2 formalizes and gives an algorithm for the second method that we propose. Another option to estimate $\tilde{\mathbf{I}}(\mathbf{x})$ consists in using a RPCA algorithm. Due to length constraints of this paper, this option is not detailed here and we refer to, e.g., [14] for a detailed explanation. Experimentally, the clique based algorithm is shown to perform better than the median and the RPCA based method in most cases, see section 4.

3.1. Median Based Algorithm

Median decision is known as a robust way to decide among samples when the noise is unknown. In our case, it would work assuming that more than 50% of pixels belong to the background. Yet, it requires a suitable generalization to deal with color images. Several choices are available to define the median value of vectorial samples. A trivial choice would be to apply a one dimensional median filter to each color channel. However, this choice leads to wrong colors. Thus, we use the median filter proposed in [22], namely

$$\arg \min_{\tilde{\mathbf{I}}(\mathbf{x}) \in \Phi(\mathbf{x})} \sum_{i=1}^{n} \left\| \mathbf{I}_i(\mathbf{x}) - \tilde{\mathbf{I}}(\mathbf{x}) \right\|_2^2, \tag{2}$$

which can be easily computed with standard algorithms [23].

3.2. Clique Based Algorithm

As we've just seen, the median based decision has limited performances due to its quite stringent assumptions. We wish to propose a new strategy to overcome these limitations. We would like to assume no specific model for the signal, the masks or the proportion of masks over background in Φ. To do so, we notice that if several images reveal the background at a given pixel, their values will be close to each other. Consequently, for each pixel $\mathbf{x} \in \Omega$, we look for a dense subset, or clique, of $\Phi(\mathbf{x})$. We define a dense clique as follows.

Definition 1. (Dense clique) *Let* $v_1, \dots, v_n \in \mathbb{R}^3$ *and* $V := \{v_1, \dots, v_n\}$. *A clique* $C \subset V$ *such that* $\operatorname{card} V = m$ *is said dense if* $\forall\, v \in C$ *its* m–1 *nearest neighbors in* V *are in* $C \backslash \{v\}$.

For every $\mathbf{x} \in \Omega$, the cliques given in definition 1, applied with $\Phi(\mathbf{x})$, can be computed using algorithm 1. As we have

Data: Set $\Phi(\mathbf{x})$ (see (1)), positive integer m
Result: Dense clique set $S(\mathbf{x})$.
Set $S = \varnothing$ and compute the $n \times m$ matrix M made with indexes of nearest neighbors (NN) of $\mathbf{I}_i(\mathbf{x})$ s.t. $\forall i \in \{1, \dots, n\}$, row$(M,i) = (i$,1st-NN $\dots, m-1$th-NN).
for *i=1,…, n* **do**
  Compute the set $E_1 := \{M(i, 1 : m)\}$
  for *j=2,…, m* **do**
    Compute the set $E_2 := \{M(M(i, j), 1 : m)\}$
    if $E_1 \neq E_2$ **then**
      Break
    end
    if *j==m* **then**
      $S := S \cup E_1$
    end
  end
end
return S

Algorithm 1: Dense clique computation.

argued, if groups of images are displaying similar values, then one of these groups can reasonably be assumed to be the background. To discriminate between these groups or cliques, we use the following definition.

Definition 2. (Meaningful clique) *We posit the same setup as in definition 1. Let* $\sigma_T > 0$ *be a given threshold. We say that a clique* C *is meaningful if* C *is the largest dense clique* C_m *such that* $\forall k \in \{2, \dots, \operatorname{card} C_m - 1\}$ *there exists dense clique* $\tilde{C}$, $\operatorname{card} \tilde{C} = k$ *and* $\operatorname{var} C_m \leqslant \sigma_T$ *or a dense clique of minimal variance such that* $\operatorname{card} C = \operatorname{card} C_m - 1$.

A clique that satisfies definition 2 can be computed by algorithm 2. Mathematically, it is possible to observe two meaningful cliques. Yet, in practice this situation never occurred during our experiments. We are now in a position to give an algorithm estimating $\tilde{\mathbf{I}}(\mathbf{x})$ given $\Phi(\mathbf{x})$: compute (2) with $C(\mathbf{x})$ obtained with algorithm 2. We now turn to the numerical evaluation of the proposed algorithm.

4. EXPERIMENTS

We compare three algorithms on real and simulated image sequences. Algorithm 2 is always applied with $\sigma_T = 15$ for images valued in $\{0, \dots, 255\}^3$. This value was found empirically using table 1 (see also subsection 4.1). The median based method consists in computing (2) on the aligned image sequence (1). The RPCA method consists in applying [24].

Data: Set $\Phi(\mathbf{x})$ (see (1)), treshold σ_T.
Result: Meaningful clique $C(\mathbf{x})$.
Set $n := \operatorname{card} \Phi(\mathbf{x})$, $m := 2$, $s := 0$, $S_{\text{pre}} := S_{\text{cur}} := \varnothing$
do
  Set $S_{\text{pre}} := S_{\text{cur}}$, $S_{\text{cur}} :=$ Algorithm 1 $(\Phi(\mathbf{x}), m)$, $m := m + 1$ and $s := \operatorname{card} S_{\text{cur}}$
while $s \geqslant 2$
Compute $\sigma^2 := \begin{cases} +\infty \text{ if } S_{\text{cur}} = \varnothing \\ \sigma^2 := \operatorname{var} C, \text{ for } C \in S_{\text{cur}} \end{cases}$
if $\sigma^2 \leqslant \sigma_T^2$ **then**
  return $C \in S_{\text{cur}}$
else
  return $\arg\min_{C \in S_{\text{pre}}} \operatorname{var} C$
end

Algorithm 2: Meaningful clique computation.

		Image 1	Image 2	Image 3	Image 4
Seq.1	Before	49.71	43.21	48.36	44.68
	After	14.48	13.32	15.05	12.18
Seq.2	Before	10.28	16.08	17.99	11.00
	After	8.57	8.02	10.47	9.85
Seq.3	Before	18.07	15.96	15.86	15.68
	After	6.14	5.61	5.18	5.47

Table 1. RMSE before/after pre-processing for 3 sequences of 4 images. RMSE after alignment is roughly below 15.

We use the implementation given in [25]. The next section discusses the acquisition protocol. More experiments can be found in [15].

4.1. Acquisition Protocol and Pre-Processing

Image sequences were acquired in a short time span, with a Canon 80D and a Sigma 30mm/f1.5 DC HSM lens. During the acquisition we ensure that the line of sight points toward the background and the focus is set on it (manually or automatically). Camera settings (ISO sensitivity, aperture and shutter speed) are manually set to avoid automatic camera adjustments. Each image is then corrected from geometric distorsions. We use a specific commercial software DxOLab [26] to perform this camera and lens dependant correction. Table 1 gives the RMSE$(I_0, I) := \frac{1}{\sqrt{\operatorname{card} \Omega}} \|I_0 - I\|_2$ after the alignment methods described in subsections 2.2 and 2.3. From table 1 we conclude that the pre-processing detailed above yields a significant decrease in terms of RMSE.

4.2. Experiments

We first give quantitative results on simulated sequences and exhibit results on real image sequences, preprocessed with the method of subsection 4.1. All results can be downloaded from the website [15].

		1	2	3	4	5
Seq.1	Clique	**91.1**	**3.2**	**0.0**	**0.0**	**0.0**
	Median	96.7	57.8	0.0	0.0	0.0
	RPCA	97.8	79.8	0.0	0.0	0.0
Seq.2	Clique		**26.0**	**0.0**	**0.0**	**0.0**
	Median		77.4	0.0	0.0	0.0
	RPCA		98.1	0.0	0.0	0.0
Seq.3	Clique		**0.0**	**2.1**	**0.0**	**0.0**
	Median		0.0	60.7	0.0	0.0
	RPCA		100	99.9	98.4	97.9

Table 2. Error rates for three noiseless simulations with clique, median and RPCA methods. The percentage of erroneous decisions as a function of the number k of observed backgrounds is given. Blue cells indicate that for sequences 2 and 3 the background is seen at least twice. These three sequences have 5 images so the median decision is correct if $k \geqslant 3$.

		1	2	3	4	5
Seq.1	Clique	**94.7**	**10.5**	8.4e-2	2.9e-3	**0.0**
	Median	97.4	61.8	**2.0e-2**	**0.0**	0.0
	RPCA	92.7	71.9	5.3e-2	4.5e-3	2.7e-3
Seq.2	Clique		**47.6**	8.5e-2	5.5e-3	**0.0**
	Median		80.3	1.7e-2	0.0	0.0
	RPCA		89.3	**0.0**	**0.0**	0.0
Seq.3	Clique		**0.0**	**6.7**	1.4e-2	**0.0**
	Median		0.0	63.3	**5.9e-3**	0.0
	RPCA		94.7	53.7	59.3	51.3

Table 3. Error rates (%). Noisy simulations with $\sigma = 5$ additive Gaussian noise. The table is organized as Table 2 ($\varepsilon := 35$). The clique method always performs better or very similarly.

4.2.1. *Simulated Experiments*

We simulated occlusion as follows. We used three background images that will be used as ground-truth. For these backgrounds, we superimposed numerically random occlusions to generate an observed sequence. We then added white Gaussian noise to these sequences. We wish to provide a quantitative comparison between the clique based method, the median based method and the RPCA method. To do so, we denote, hereinafter, by I_0 the ground-truth and $\tilde{I}$ the estimated background. The pixels where the background is observed $k \in \{1, \ldots, n\}$ times, in a sequence of n images, are given by the set

$$T(k) := \{\mathbf{x} \in \Omega : \varphi(\mathbf{x}) = k\}, \tag{3}$$

where $\varphi(\mathbf{x}) := \text{card } \{i \in \{1, \ldots, n\} : \|I_i(\mathbf{x}) - I_0(\mathbf{x})\|_2 < \varepsilon\}$ and $\varepsilon \geqslant 0$ is some threshold. For $k \in \{1, \ldots, n\}$, the error rate is defined by

$$R(k) := \frac{\text{card } \left\{\left\|I_0(\mathbf{x}) - \tilde{I}(\mathbf{x})\right\|_2 < \varepsilon\right\}}{\text{card } T(k)}. \tag{4}$$

Tables 2-3 give the error rates for four simulated sequences in the noiseless and noisy cases. In these experiments the clique method based on algorithm 2 always performs better.

4.2.2. *Experiments with Real Sequences*

We give comparative results on real image sequences of our own in figure 2. We recall that the experimental protocol and pre-processing is given in subsection 4.1. We notice, at various positions where the background is occluded several times, that the clique based method performs better than the median or the RPCA methods. We recall that an implementation and more experiments, on which similar conclusions can be drawn, can be found in [15].

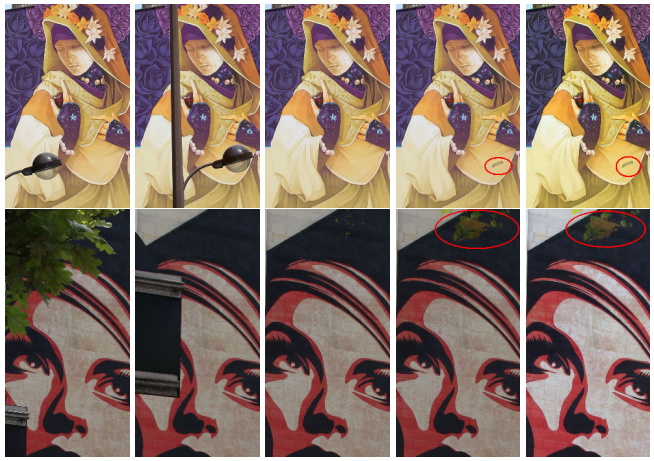

Fig. 2. Real sequences. From left to right: two aligned frames, the clique, the median and the RPCA method. The red circle enlights defects in median and RPCA results.

5. CONCLUSION

A new algorithm for occlusion detection and restoration was proposed. The algorithm is based on a temporal non-linear filter that relies on a meaningful clique computation. This new algorithm is fast, simple and robust. This algorithm was demonstrated to compare advantageously with a color median, as well as with a much more sophisticated method, RPCA. Notably, no assumption was made on the occlusion shapes, textures, colors or motions. A future work could generalize the approach to a spatio-temporal filtering method.

6. REFERENCES

[1] K. Garg and S. K. Nayar, "Detection and removal of rain from videos," in *Computer Vision and Pattern Recognition, 2004, CVPR 2004.* IEEE, 2004, vol. 1, pp. I–I.

[2] X. Zhang, H. Li, Y. Qi, W. K. Leow, and T. K. Ng, "Rain removal in video by combining temporal and chromatic properties," in *Multimedia and Expo, 2006*. IEEE, 2006, pp. 461–464.

[3] P. C. Barnum, S. Narasimhan, and T. Kanade, "Analysis of rain and snow in frequency space," *International journal of computer vision*, vol. 86, no. 2-3, pp. 256, 2010.

[4] A. Yamashita, F. Tsurumi, T. Kaneko, and H. Asama, "Automatic removal of foreground occluder from multifocus images," in *Robotics and Automation (ICRA), IEEE Int. Conf.* IEEE, 2012, pp. 5410–5416.

[5] T. Xue, M. Rubinstein, C. Liu, and W. T. Freeman, "A computational approach for obstruction-free photography," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 4, pp. 79, 2015.

[6] R. Wan, B. Shi, T. A. Hwee, and A. C. Kot, "Depth of field guided reflection removal," in *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 21–25.

[7] C. Sun, S. Liu, T. Yang, B. Zeng, Z. Wang, and G. Liu, "Automatic reflection removal using gradient intensity and motion cues," in *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 2016, pp. 466–470.

[8] M. Park, K. Brocklehurst, R. T. Collins, and Y. Liu, "Image de-fencing revisited," in *Asian Conference on Computer Vision*. Springer, 2010, pp. 422–434.

[9] Y. Mu, W. Liu, and S. Yan, "Video de-fencing," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 7, pp. 1111–1121, 2014.

[10] Y. Zhang, J. Xiao, and M. Shah, "Motion layer based object removal in videos," in *Application of Computer Vision, 2005. WACV/MOTIONS'05. 7 IEEE Workshops*. IEEE, 2005, vol. 1, pp. 516–521.

[11] V. Vaish, B. Wilburn, N. Joshi, and M. Levoy, "Using plane+ parallax for calibrating dense camera arrays," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proc. 2004 IEEE Comp. Soc. Conf.* IEEE, 2004, vol. 1, pp. I–I.

[12] A. Criminisi, P. Pérez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Transactions on image processing*, vol. 13, no. 9, pp. 1200–1212, 2004.

[13] A. Newson, A. Almansa, Y. Gousseau, and P. Pérez, "Non-local patch-based image inpainting," *Image Processing On Line*, vol. 7, pp. 373–385, 2017.

[14] T. Bouwmans, N. S. Aybat, and E-H. Zahzah, *Handbook of Robust Low-Rank and Sparse Matrix Decomposition: Applications in Image and Video Processing*, Chapman & Hall/CRC, 2016.

[15] "Occlusion Removal web page," `https://perso.telecom-paristech.fr/xiayang/occlusion_removal/`.

[16] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*, Cambridge university press, 2003.

[17] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," in *Readings in computer vision*, pp. 726–740. Elsevier, 1987.

[18] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*. IEEE, 1999, vol. 2, pp. 1150–1157.

[19] F. Pitié, A. C. Kokaram, and R. Dahyot, "Automated colour grading using colour distribution transfer," *Computer Vision and Image Understanding*, vol. 107, no. 1-2, pp. 123–137, 2007.

[20] R. Ramanath, W. E. Snyder, Y. Yoo, and M. S. Drew, "Color image processing pipeline," *IEEE Signal Processing Magazine*, vol. 22, no. 1, pp. 34–43, 2005.

[21] R. Nguyen, D. K. Prasad, and M. S. Brown, "Raw-to-raw: Mapping between image sensor color responses," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3398–3405.

[22] J. Astola, P. Haavisto, and Y. Neuvo, "Vector median filters," *Proceedings of the IEEE*, vol. 78, no. 4, pp. 678–689, 1990.

[23] C. A. R. Hoare, "Quicksort," *The Computer Journal*, vol. 5, no. 1, pp. 10–16, 1962.

[24] S. Hauberg, A. Feragen, and M. J. Black, "Grassmann averages for scalable robust PCA," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3810–3817.

[25] A. Sobral, T. Bouwmans, and E.-H. Zahzah, "LRSLibrary: Low-Rank and Sparse tools for Background Modeling and Subtraction in Videos," in *Robust Low-Rank and Sparse Matrix Decomposition: Applications in Image and Video Processing*. CRC Press, Taylor and Francis Group., 2015.

[26] DxO, "DxO Optics Pro 11," 2016.