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 occur in image sequences. In contrast to many methods of the literature, no assumption is made on occlusions 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 show that this new clique-based method provides best performances 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 scenic view many people experience the difficulty that someone wander into the shot they wish to take. 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 combing several photographs.

Several works have yet addressed this problem. Some authors focus on specific kinds of obstacles as, for instance, raindrops [1, 2, 3, 4, 5], grids [6] or fences [7]. Some authors use a very dense sequence of images so that they can derive an optical flow from which a depth map is deduced and the farthest away surface is kept [8]. Alternately, some authors rely on specific dense sensor configurations that allow for a statistical decision [9]. If no information on the underlying scene is observed, an ultimate strategy consists in inpainting missing areas with the most probable content [10, 11] after a mask detection scheme is performed [6]. Yet, inpainting strategies are prone to errors, i.e., mis-estimated 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 masks shapes don't hold true. For all of theses reasons, we believe that a simple, fast and reliable algorithm should be proposed.

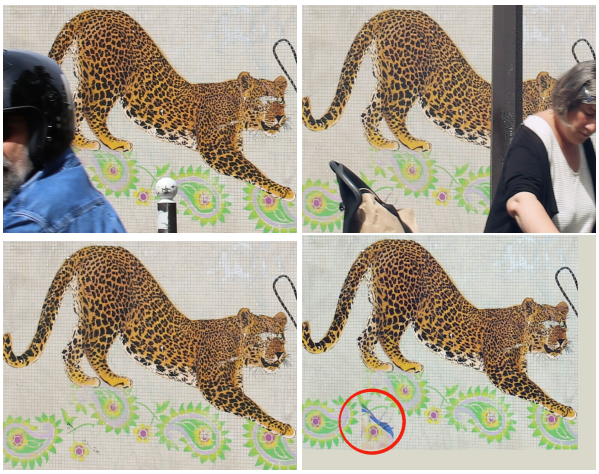

Fig. 1. On top: two frames of an image sequence after geometric and photometric alignment. Bottom left (resp. right) the proposed solution (resp. right) the RPCA method. The red circle en-lights a defect in the RCPA reconstruction.

Roughly, the solution proposed in this paper relies on the combined motions of the photographer and of the masks to ensure that the sequence gives the entire background. 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 shapes, color, motion or textures for the masks or obstacles. 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 [12].

Outline of the paper: Section 2 gives the setup considered in this paper and details the methods we use to align geometrically and photometrically image sequences. Section 3 mainly discusses a new method for background detection based on a meaningful clique detection. Section 4 gives, for real and simulated sequences, 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 and directions of our current work. *A webpage with an implementation, image sequences and numerical results is available [13].*

2. PROPOSED METHOD

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

2.1. Framework and Assumptions

As we've seen in section 1, many methods in the literature consider specific kind 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 assumption 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 differences when looking from different positions. A limitation of the above assumptions is that the algorithm we propose in this paper cannot be expected to work well when observing backgrounds with reflecting surfaces or specular reflectors like mirrors. We expect that lighting conditions are almost constant during the acquisition.

2.2. Geometrical Image Alignment

The approach we employ is straightforward. A reference frame is chosen. The assumption 2) allows us re-sample the observed frames on the reference frame using an homography [14] and bi-linear interpolation. The homography parameters are computed using RANSAC [15] on SIFT matches [16][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 experiment differences that depends on the many uncontrolled differences between images. In addition, the camera white-balance algorithm also tends to modify the color content between images. The observed color distributions depends not only on the background but also on the masks or occlusions. Thus, we can't use standard color transfer algorithms [17] to equalize the images. To solve the problem, we determine color transfer maps between images. Indeed, digital camera conversion from input intensities to output vectors can be approximated by an invertible function [18]. Following [19], we use an order two polynomial model to compute this color transfer mapping. As we shall see, the use of an order 2 polynomial model often gives good results. Experiments show a poor correlation between channels so we can compute the mapping for each color channel independenty. The polynomial coefficients are computed from three pairs of SIFT matches. We use pixels corresponding to SIFT matches obtained from the geometric alignment then apply a RANSAC strategy to robustify the selection. The computed polynomial is applied to the whole image and the background has almost constant colors in the entire sequence.

[1] A SIFT match is defined, as usual, by 1st neighbor $\leqslant 15 \times$ 2nd neighbor.

3. IMAGE RECONSTRUCTION

After geometric and photometric alignments described in sections 2.2 and 2.3, we obtain a stack of photometrically and geometrically 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 (section 3.1) criterion. Section 3.2 formalizes and gives algorithms to compute the second method that was developed for this project. Another option to estimate $\tilde{\mathbf{I}}(\mathbf{x})$ consists in using a robust RPCA algorithm. Due to the severe length constraints of this paper, this option is not detailed here and we refer to, e.g., [12] 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 the pixels belong to the background and provided a suitable generalization is used to deal with color images. Several choices exist 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 lead to wrong colors in our experiments. Thus, we use the median filter proposed in [20], namely

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

that can be easily computed with standard algorithms [21].

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 at a given pixel several images are displaying the background, then these values will be close. Consequently, for each pixel $\mathbf{x} \in \Omega$, we look for a dense subset, or clique, of $\Phi(\mathbf{x})$. To do so, 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've 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.

Data: Set $\Phi(\mathbf{x})$ (see (1)), positive integer m
Result: Meaningful cliques set $S(\mathbf{x})$.
Set $S = \varnothing$ and compute the $n \times m$ matrix made with indexes of nearest neighbors (NN) of $\mathbf{I}_i(\mathbf{x})$. Namely $\forall i \in \{1, \dots, n\}$, $\operatorname{col}(M, i) = (i, 1\text{st-NN} \dots, m\text{–1th-NN})$.
for *i=1,. . . , n-m+1* **do**
  for *j=0,. . . ,m-1* **do**
    if $col(M, i) \neq col(M, i + j)$ **then**
      Break
    end
    if *j==m-1* **then**
      $S := S \cup \operatorname{col}(M, i)$
    end
  end
end
return S
Algorithm 1: Dense clique computation.

Definition 2. (Meaningful clique) *We posit the same setup as in definition 1. We say that a dense clique C is meaningful if every other dense clique $\widetilde{C}$ satisfy* $\operatorname{card} \widetilde{C} \leqslant \operatorname{card} C$ *and* $\operatorname{var} C \leqslant \sigma_T^2$*, where σ_T^2 is a given threshold.*

A clique that satisfy 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 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.

Data: Set $\Phi(\mathbf{x})$ (see (1)), treshold σ_T^2.
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.

4. EXPERIMENTS

We compare three algorithms on real and simulated images. 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 [22]. We use the implementation given in [23]. The next section discusses the acquisition protocol for the real image sequences. More experiments can be found in [13].

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 image acquisition, the background of interest is maintained in the center of the field and the focusing is done on it, either manually or automatically. Camera settings (ISO sensitivity, aperture and shutter speed) are manually controlled to avoid various effects which may appear with automatic camera settings when changing the field of view or because of different masks. Then, each image is carefully corrected from camera defects: geometric distorsions, chromatic aberrations, vignetting effects. These elements proved to be important to improve color matching at pixel level after image alignment. We also prevent the automatic white balance correction which may be different image by image depending on the scene content. To perform these camera and lens dependant corrections, we make use of a specific commercial software DxO-Lab [24]. This pre-processing yields to a significant decrease in images disparity, after the alignment methods described in sections 2.2 and 2.3. This pre-processing yields to a significant decrease in images disparity, see table 1 that gives the $\text{RMSE}(I_0, I) := \frac{1}{\sqrt{\operatorname{card} \Omega}} \|I_0 - I\|_2$.

4.2. Experiments

We sequentially give quantitative results on simulated sequences then on real image sequences preprocessed with the method of subsection 4.1. For the simulated experiments, we consider only the clique based algorithm developed in this pa-

		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 are roughly below 15.

per and the median based method. Indeed, the RPCA method produces images up to an affine contrast change. Therefore, quantitative results cannot be computed for the RPCA method. Yet, RPCA is considered in the real experiments where we only compare the visual quality of the estimated backgrounds.

4.2.1. Simulated Experiments

We simulated occlusion as follows. We used four background images that will be used as ground-truth. For each of these background, we superimposed numerically occlusions randomly 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 and the median based 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 $k \in \{1, \dots, n\}$ times, in a sequence of n images, are given by the set

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

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

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

Tables 2-5 give the error rate 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 images sequences of our own in figure 2. We recall that the experimental protocol and pre-processing is given in subsection 4.1. We notice that the clique based method performs better or similarly than the median or the RPCA methods. We recall that an implementation and more experiments can be found in [13].

5. CONCLUSION

A new algorithm for occlusion detection and restoration was proposed. The algorithm is a temporal non-linear filter that re-

Dragon	1 (1e3)	2 (1e4)	3 (4e4)	4 (6e4)	5 (2e5)
Clique	**91.1**	**3.2**	0.0	0.0	0.0
Median	96.7	57.8	0.0	0.0	0.0
Leopard	1 (0)	2 (1e3)	3 (2e4)	4 (5e4)	5 (5e5)
Clique	$\varnothing$	**26.0**	0.0	0.0	0.0
Median	$\varnothing$	77.4	0.0	0.0	0.0
Jungle	1 (0)	2 (7e3)	3 (3e4)	4 (7e4)	5 (8e5)
Clique	$\varnothing$	0.0	**2.1**	0.0	0.0
Median	$\varnothing$	0.0	60.7	0.0	0.0
Bakery	1 (0)	2 (1e4)	3 (2e4)	4 (1e5)	5 (8e5)
Clique	$\varnothing$	0.0	**0.0**	0.0	0.0
Median	$\varnothing$	0.0	47.7	0.0	0.0

Table 2. Error rates. Noiseless simulations. The first column gives the sequence name and the algorithm considered: the clique (algorithm 2) or the median methods (2). The second column gives the error rate (4) ($\varepsilon := 3$) for $k = 1$, the number in parentheses is $\text{card}\,M(k)$. The second row gives the error rate for algorithm 2. The other columns are organized similarly for different value of $k \in \{1, \dots, n\}$. The clique method always performs better.

		5	4	3	2	1
1	Clique	**91.1**	**3.2**			
	Median	96.7	57.8			
2	Clique		**26.0**			
	Median		77.4			
3	Clique					
	Median			47.7		

Table 3. Error rates with 3 noiseless simulations and with clique or median decision. The percentage of erroneous decisions as a function of the number n of masks is presented. The blue cells are when there is no pixel hidden by n masks. The pink cells are those where no error is made. For these 3 cases the number of images in a sequence is 6 so that the median value is sure whenever $n < 3$.

Dragon	1 (1e3)	2 (1e4)	3 (4e4)	4 (6e4)	5 (2e5)
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
Leopard	1 (0)	2 (1e3)	3 (2e4)	4 (5e4)	5 (5e5)
Clique	$\varnothing$	**47.6**	8.5e-2	5.5e-3	0.0
Median	$\varnothing$	80.3	**1.7e-2**	**0.0**	0.0
Jungle	1 (0)	2 (7e3)	3 (3e4)	4(7e4)	5(8e5)
Clique	$\varnothing$	0.0	**6.7**	1.4e-2	0.0
Median	$\varnothing$	0.0	63.3	**5.9e-3**	0.0
Bakery	1 (1)	2 (1e4)	3 (2e4)	4 (9e4)	5 (8e5)
Clique	100	**1.0**	1.2e-2	1.1e-3	0.0
Median	100	52.9	0.0	**0.0**	**0.0**

Table 4. Error rates. Noisy simulations: $\sigma = 5$ additive Gaussian noise. The table is organized as Table 2 ($\varepsilon := 35$). The clique method always performs better or very similarly.

		1	2	3	4	5
1	Clique	**94.7**	**10.5**	0.084	0.003	
	Median	97.4	61.8	**0.02**		
2	Clique		**47.6**	0.085	0.005	
	Median		80.3	**0.017**		
3	Clique			**6.7**	0.014	
	Median			63.3	**0.006**	
4	Clique	100	**1.0**	0.012	0.001	
	Median	100	52.9			

Table 5. Error rates in case of 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.

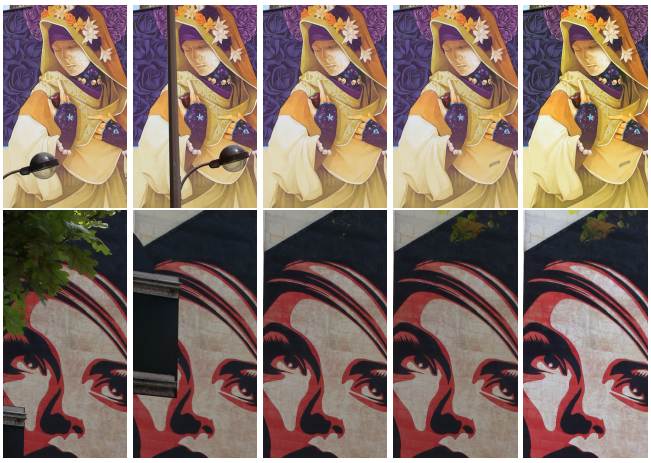

Fig. 2. Real experiments. From left to right: two aligned frames, the clique, the median and the RPCA method.

lies on a meaningful clique computation. This new algorithm is fast, simple and robust. This algorithm was demonstrated to compare advantageously with a much more sophisticated method such as a RPCA. Notably, no assumption was made on the occlusions 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. Proceedings of the 2004 IEEE Computer Society Conference on*. IEEE, 2004, vol. 1, pp. I–I.

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

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] Yunjun Zhang, Jiangjian Xiao, and Mubarak 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.

[9] Vaibhav Vaish, Bennett Wilburn, Neel Joshi, and Marc 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.

[10] 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.

[11] 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.

[12] 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.

[13] "Object Removal web page," `http://perso.telecom-paristech.fr/~xiayang/object_removal/`.

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

[15] 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.

[16] 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.

[17] 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.

[18] 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.

[19] 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.

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

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

[22] 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.

[23] 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.

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