CORE: A COnfusion REduction Algorithm for Keypoints Filtering

Emilien Royer, Thibault Lelore and Frédéric Bouchara
Université de Toulon, CNRS, LSIS UMR 7296, 83957 La Garde, France

Keywords: Keypoints Filtering, Computer Vision, Feature Matching, Kernel Density Estimator.

Abstract: In computer vision, extracting keypoints and computing associated features is the first step for many applications such as object recognition, image indexation, super-resolution or stereo-vision. In many cases, in order to achieve good results, pre or post-processing are almost mandatory steps. In this paper, we propose a generic pre-filtering method for floating point based descriptors which address the confusion problem due to repetitive patterns. We sort keypoints by their unicity without taking into account any visual element but the feature vectors's statistical properties thanks to a kernel density estimation approach. Even if highly reduced in number, results show that keypoints subsets extracted are still relevant and our algorithm can be combined with classical post-processing methods.

1 INTRODUCTION

Over the last recent years, keypoint detection and feature computation have seen an increasing attention in computer vision researches, partly thanks to the ongoing developpment of robotic and need of efficient image databases queries. As a major contribution we can cite the SIFT (Lowe, 1999) descriptor by D. Lowe. Based on oriented gradient histograms, it proved to be very efficient (Mikolajczyk and Schmid, 2005) and inspired many others such as ASIFT (Morel and Yu, 2009) and SURF (Bay et al., 2006). Nowadays it is still used in modern applications and has even been ported to GPU architectures (Wu, 2007). However, its computation times are not suitable for real-time applications and the rises of small embedded platforms such as smartphones aspired to faster computation times and less memory consumption. Thus, in 2010 Calonder *et al.* introduced the BRIEF (Calonder et al., 2010) descripor, leading the way to the binary descriptors field which produced ORB (Rublee et al., 2011), BRISK (Leutenegger et al., 2011), FREAK (Ortiz, 2012), D-BRIEF (Trzcinski and Lepetit, 2012) and state-of-the-art Bin-boost (T. Trzcinski and Lepetit, 2013). In this area, some descriptors propose a way of improving keypoint selection. For example, ORB orders the FAST (Rosten and Drummond, 2006) responses by a harris corner measure (Harris and Stephens, 1988). With our contribution, we propose a solution to both generally improve the selection and to address a specific case that we are presenting in the next section.

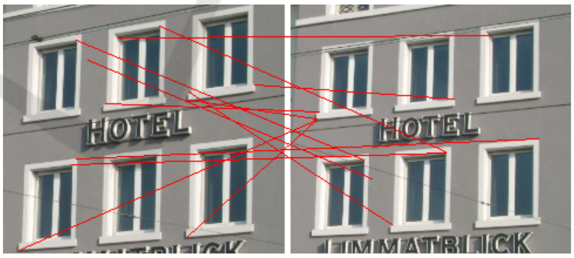

Figure 1: Example from the Zurich Building Image Database of repetitive patterns leading in ”good-false” matches with the sift descriptor.

1.1 The Repetitive Patterns Problem

A frequent and troublesome problem easily encountered when trying to match pairs in different images is the repetitive pattern case, as we can see in figure 1: the exact same pattern is present in multiple occurencies within the image. These visual features make it highly responsive to saliency analysis, returning numerous keypoints that have almost the same feature vectors, which results in high confusion during matching phase. Usually, the mismatch problem is handled from a given putative point correspondences by different kinds of approaches. A first kind of methods is based on a robust statistic estimation such as LMS (Least Median of Squares) or M-estimators. In (Deriche et al., 1994) Deriche *et al.* applied the LMS for the robust estimation of the fundamental matrix. In a similar approach Torr *et al* (Torr and Murray, 1995) proposed a method for the estimation of both

Royer E., Lelore T. and Bouchara F..
CORE: A COnfusion REduction Algorithm for Keypoints Filtering.
DOI: 10.5220/0005309405610568
In *Proceedings of the 10th International Conference on Computer Vision Theory and Applications* (VISAPP-2015), pages 561-568
ISBN: 978-989-758-089-5

the fundamental matrix and motion estimation. Another robust estimation methods can be found in the literature such as the algorithms proposed by Ma *et al* (Zhao et al., 2011; Ma et al., 2014).

Another kind of methods, known as resampling methods, act by trying to get a minimum subset of mismatch-free correspondence. Methods belonging to this category are usually extensions of the well known RANSAC (RANdom SAmple Consensus) (Fischler and Bolles, 1981) such as MLESAC (Torr and Zisserman, 2000) or SCRAMSAC (Sattler et al., 2009). We can also cite (Pang et al., 2014) and (Rabin et al., 2007).

Other algorithms are based on different approaches as the ICF (Identifying point correspondences by Correspondence Function) proposed by Li *et al* (Li and Hu, 2010).
Another way to consider the mismatch problem is to filter out repetitive patterns in each image. Such a priori approaches may be combined with the previous methods that are performed a posteriori from a given putative point correspondences. When looking at the literature, detecting repetitive pattern is a known issue in several different applications although it is reputed to be difficult. Repetitive structures can be detected through symmetry analysis (Loy and Eklundh, 2006; Lee et al., 2008; Liu et al., 2004) and despite being mostly 2D analysis, recent propositions try to take into account non-planar 3D repetitive elements (Jiang et al., 2011; Pauly et al., 2008). Mortensen *et al.* enrich the SIFT descriptor with information about the image global context (Mortensen et al., 2005), inspired by shape contexts (Belongie et al., 2002). The SERP (Mok et al., 2011) descriptor and the CAKE (Martins et al., 2012) keypoint extractor both rely on kernel density estimation (Parzen, 1962). The first one uses mean-shift clustering on SURF descriptors, whereas the second one builds a new keypoint extractor based on shannon's definition of information. As we're about to see in the next section, our approach does also rely on kernel density estimation but in a different way.

In this paper we propose a new approach to cope with the keypoints confusion problem. We don't take into account the keypoints visual properties since they may vary with the type of extractor chosen, but instead we analyse the statistics properties of their associated feature vectors. We estimate a numerical value that is associated to the confusion risk of a given feature vector between another vector in a different image. With this criterion, we can then sort the keypoints from low confusion risk, to high confusion risk. With the right threshold, we can thus decide which points should be discarded and which ones should be kept. The rest of the paper is organized as follow: Section 2.1 will present an overwiew of our proposed method. In section 2.2 we will explain the criterion computation. Section 2.3 will address the problem of threshold setting. Finally, Section 3.1 will detail our experiments methodology and sections 3.2 and 4 will respectively present results and conclusions.

Further in the text we will use the following notation: we let $P_x(y)$ be the probability $\Pr(x = y)$ that the variable x is equal to the value y.

2 PROPOSED METHOD

2.1 Overview

Let I be the image resulting of the observation (with a camera) of a specific scene. Let I' be (a potential) another observation of the same scene in which changes result from various transformations such as perspective changes, light modifications, etc. In our model, I is deterministic whereas I' is a potential (not yet observed) different version of I and is hence considered to be stochastic. Let now $u_{i,\ i\in\{1,\dots,N\}}$ be D-dimensional feature vectors computed on N keypoints of I and let $u'_{i,\ i\in\{1,\dots,N\}}$, be their N respective equivalents in I'. We assume that even if descriptors try to be invariant as much as possible to most transformations, each feature vector in image I is subject to slight variations in image I' we can assimilate as randomness. By doing so we consider u'_i as random vectors and we shall define a criterion associated to each keypoints of I that characterizes the confusion risk, i.e. a value correlated to the probability that in I', a vector $u'_{j,\ j\neq i}$ is closer to u_i than u_j.
For each keypoint i of I we define C_i, the criterion, as the probability density that any other random $u'_{j,\ j\neq i}$ is equal to u_i, *i.e.* $P_{u'_{j,\ j\neq i}}(u_i)$. This density should act as a criterion for separating relevant and high confusion risk keypoints.

From this definition, we can write:

$$C_i \equiv P_{u'_{j,\ j\neq i}}(u_i) = \sum_{j\neq i} \Pr(k = j, u = u_i) \quad (1)$$

$$= \sum_{j\neq i} P_{k,\ k\neq i}(j) P_{u/j}(u_i) \quad (2)$$

where $P_{k,\ k\neq i}(j)$ denotes the probability of choosing keypoint j and $P_{u/j}(.)$ is the probability density function (PDF) of the feature vector given the keypoint number. We simply assume $P_{k,\ k\neq i}(j) = \frac{1}{N-1}$ (the $N-1$ keypoints are equiprobable) and we note $P_{u/j}(u) = \frac{1}{h} K\left(\frac{|u-u_j|}{h}\right)$ where K is a normalized symmetric function and h is a smoothing parameter. We thus obtain the estimation of C_i by the classical Parzen-Rosenblatt kernel density estimator (KDE):

$$C_i = \frac{1}{h(N-1)} \sum_{j \neq i} K\left(\frac{|u_i - u_j|}{h}\right) \quad (3)$$

The CORE algorithm (given in Algorithm 1) is very straightforward and easy to implement.

Algorithm 1: CORE algorithm.

Data: I : image input
Data: p : probability confusion tolerated
Data: D : descriptor dimension
Data: σ : average variance of descriptor's feature vectors
Data: $C_{th} \leftarrow$ findThreshold(p, σ, D) *(b)*
Result: χ : keypoint set returned
$K \leftarrow$ keypoint set detected
$U \leftarrow$ associated feature vectors
for $u_i \in U$ **do**
  $c_i \leftarrow$ KDE(u_i, U) *(a)*
end
for $k_i \in K$ **do**
  if $c_i < C_{th}$ **then**
    Add k_i to χ
  end
end
return χ

Steps *(a)* and *(b)* are explained in next subsections.

2.2 Criterion Computation

We suppose that the vectors variations causes are numerous and are either from natural origins or can be considered as such. Therefore, it makes sense to consider this behavior to be Gaussian. With this assumption we can define K as the classical Gaussian Kernel:

$$K(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}u^2\right) \quad (4)$$

From such a definition, h takes the meaning of a standard deviation σ of which we further address the setting in our experiments section.
Thus, the criterion formula is :

$$C_i = \frac{1}{(N-1)\sigma\sqrt{2\pi}} \sum_{i \neq j} \exp\left(-\frac{d(u_i,u_j)^2}{2\sigma^2}\right) \quad (5)$$

where $d(u_i,u_j) = \sqrt{\|u_i - u_j\|}$ is the euclidean distance between vector u_i and u_j.

2.3 Thresholding

Again, a relevant value of the threshold C_{th} to apply on the $C_{i,\ i \in \{1,\dots,N\}}$ can be estimated by considering the confusion problem with a probabilistic point of view. With the notations of the previous section, let u_i and u'_i be the feature vectors computed on the same keypoint i of two different versions of a scene. Let now $v_i = u'_i - u_i$, $v_j = u'_j - u_i$, $d_i^2 = \|v_i\|^2$ and $d_j^2 = \|v_j\|^2$ where u_j, u'_j are the corresponding feature vectors computed on another keypoint j.

To estimate C_{th} we shall express C_i as a function of $p = \Pr(d_j^2 < d_i^2)$ the probability of a confusion. In our approach, p is a user-defined parameter which tunes an acceptable confusion rate. To derive this relation we need first to estimate $P_{d_j^2}(.)$, (and hence $P_{v_j}(.) = P_{u'_{j,\ j \neq i}}(.) - u_i$) which is governed by the distribution of the $u_{j,\ j \neq i}$. However, we shall assume that p only depends on the behavior of $P_{v_j}(.)$ in a small neaborhood of u_i. We hence approximate $P_{v_j}(.)$ by a D-dimensional uncorrelated Gaussian distribution $N(.;0,\Sigma_{v_j})$ of which the central value $\Pr(v_j = 0) = P_{v_j}(0) = C_i$ by vertu of the definition of C_i given in the previous section. The diagonal element σ_{v_j} of the covariance matrix Σ_{v_j} is simply related to C_i by considering the normalisation condition on $P_{v_j}(.)$ which can be written:

$$C_i = (2\pi\sigma_{v_j}^2)^{-D/2} \quad (6)$$

From this assumption, $P_{d_j^2}(.)$ is given by a chi-squared distribution with D degrees of freedom which can be approximated by a Gaussian law $N(.;E_j,\sigma_j)$ due to the large value of D. The values of E_j and σ_j are classically related to the values of σ_{v_j} and D by: $E_j = \sigma_{v_j}^2 D$ and $\sigma_j = \sigma_{v_j}^2\sqrt{2D}$.

Thanks to the Gaussian assumption on the u'_i values and using the same considerations as before, we can also approximate $P_{d_i^2}$ by a gaussian law $N(.;E_i,\sigma_i)$ with $E_i = \sigma^2 D$ and $\sigma_i = \sigma^2\sqrt{2D}$.
From these definitions we can now write:

$$\begin{aligned} p &= \Pr(d_j^2 < d_i^2) && (7)\\ &= \int_{-\infty}^{\infty}\int_{x}^{\infty} P_{d_j^2}(x) P_{d_i^2}(y)\,dy\,dx && (8)\\ &= \int_{-\infty}^{\infty}\int_{x}^{\infty} N(x;E_j,\sigma_j) N(y;E_i,\sigma_i)\,dy\,dx && (9) \end{aligned}$$

$$= \frac{1}{2} - \frac{1}{2\sigma_j\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp\left[\frac{-(x-E_j)^2}{2\sigma_j^2}\right] \times$$
$$\mathrm{erf}\left[\frac{x-E_i}{\sigma_i\sqrt{2}}\right] dx \qquad (10)$$
$$= \frac{1}{2}\left[1+\mathrm{erf}\left(\frac{E_i-E_j}{\sqrt{2(\sigma_i^2+\sigma_j^2)}}\right)\right] \qquad (11)$$

After a straightforward, albeit a bit tedious, calculation we obtain from (11):

$$\sigma_{v_j}^2 = \sigma^2 \frac{D+2\sqrt{\gamma(D-\gamma)}}{D-2\gamma} \qquad (12)$$
$$\text{with} \qquad \gamma = 2\,\mathrm{erf}^{-1}(2p-1)^2 \qquad (13)$$

From (12) and (13), the threshold C_{th} which corresponds to a specific p is then given by (6).

3 EXPERIMENTS

3.1 Protocol

A quick example of keypoints filtered by proposed method with the SIFT descriptor is shown with figure 2 where the confusion probability tolerated is 10%. As we can see, the vast majority of the chessboard image's points are removed except for some on the corners, whereas the ones on the photograph are mostly kept. This tends to confirm the wanted behavior of our algorithm.

For validating our contribution, we're looking to prove that our algorithm does actually extract a better keypoint subset less subject to confusion. For this, we choose the classical application which consists in matching keypoints pairs in different images. Specificaly, we use a similar approach as used by SCRAMSAC by estimating an underlying model (*i.e.* fundamental matrix) and analysing the ratio of correspondences consistent with it, called the inliers. We first apply our experiments on a personnal set of 10 couples of document images captured by a smartphone camera. Printed document images are very good candidates for confusion reduction due to the letters and words repetitions. Moreover, their visual properties make them highly responsive to saliency analysis, resulting in a profusion of keypoints returned; usually around 30.000 for a 2560x1920 picture with default sift parameters. Thus, we also test our method as a way of reducing huge keypoint sets without relying on visual analysis. As for the descriptor selection, considering its wide popularity and effiency, it

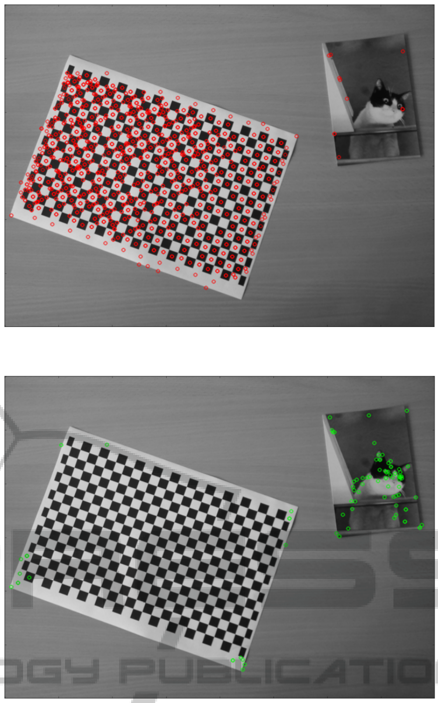

Figure 2: CORE application, top shows keypoints removed and bottom is keypoints kept with $p = 0.1$.

is an obvious choice to base our protocol experiment on the SIFT one. We also follow the idea of David Lowe in (Lowe, 1999) to keep only high-quality feature matches: we reject poor matches by computing the ratio between the best and second-best match (labelled 2NN for 2 nearest neighbours). If the ratio is below a given threshold (we use 0.8), the match is discarded as being low-quality.

We proceed as follows: for each image pair, we apply our CORE algorithm on the keypoints returned by SIFT. This returns a reduced keypoint set with which we establish correspondences by brute-force matching. We then use the RANSAC algorithm to estimate the fundamental matrix and analyse the inlier ratio. For a fair comparison, we do the same with another keypoint subset by following Lowe idea of saliency analysis by a contrast threshold, so we end up with a different keypoint set with equal size. On both of these approaches, we also apply the SCRAMSAC test to see how his matching filter behaves with these two different pre-processing methods. Last, to serve as a control test we extract a random keypoint subset with same size in order to prove that our method (as well as Lowe's one) is better and makes more sense than randomness. We repeat this for different p values, respectively 0.5, 0.25, 0.15, 0.10 and 0.05.

Finally, a valid criticism would be that analysing the inlier ratio might not be always pertinent since the fundamental matrix computed is not always accurate. That's why we propose a manual validation step: for each couple of images, we apply the sift algorithm on both images to detect and compute feature vectors. From here, we build a first set of results by brute-force matching the vectors; this will serve as a base for comparisons. Thereafter, we apply the 2NN filter; this is our second result-set. Last, we use our CORE algorithm in order to remove keypoints that could lead to confusion before applying the matching and previous filter, giving us our third and last result-set. For each of these three sets, an operator manually evaluates each match, giving us the table in the result section 3.2. We also use the Zurich Image database (Shao and Gool, 2003) instead of document images to show that our algorithm is not exclusive to these and arbitrarly set a fixed p value of 0.1. Plus, since we're not computing average results here we also include two image couples from images figure 2.

But before presenting our results, we still need to address the σ setting as shown earlier in section 2.2: since it characterizes the feature vector values variations, we use our images set to compute the global mean value of variances of vectors elements thanks to the correct matches manually checked. We found it to be roughly around 32.135 for the SIFT descriptor. However, even if early tests didn't find notable sensibility for values above 10, for very specific applications it could be understandable to re-evaluate it more precisely.

3.2 Results

Results from first part of the experiment are presented with figure 3. We see that for every p values, number of inliers is always greater than other subsets of equal size resulting from saliency analysis. Moreover, with small p values (between 0.25 and 0.05), inlier ratio is always improved by CORE pre-processing and starting with $p = 0.15$, CORE is doing better than SCRAMSAC.

However, for $p = 0.5$ (50% of confusion tolerated), the inlier ratio is actually smaller with our method. This could comes from the large confusion tolerated that doesn't remove enough keypoints: we don't take advantage of confusion reduction and some very similar keypoints where removed whereas their feature vector transformation may have not been enough to generate confusion. So we recommend using p values being inferior than 0.25 and best results seem to be achieved with 0.10. Not studied here, another advantage of our algorithm would be the speed-up gained during matching phase and model estimation as we observed the average computation time to

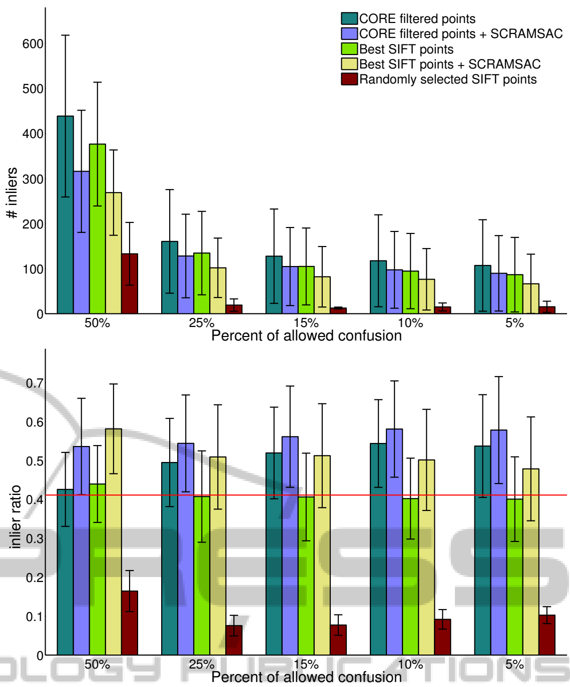

Figure 3: Average results of first part of the test with differents filters. For each p value, we compare the results with subsets of equal size. Top: raw numbers of inliers, bottom: inlier ratio. Horizontal red line corresponds to SIFT inlier ratio without any filtering.

be 20 times faster than without filtering. Finally, it is worth noting that our pre-processing filter (CORE) behaves well with post-processing (SCRAMSAC) by always increasing the inlier ratio, regardless of the p value used and the poor results from control test based on randomness prove the relevance of pre-processing.

Now, concerning the second part of the test, shown by table 1, we can see that our contribution globally improves the good matching ratio: we find a mean increasing value of 8.52% for the Zurich images. Images 4.c and 4.i show slight improvements (with respectively 1.13% and 2.72% ratio increasing) while the other ones extracted from this dataset range from 6.22% to 13.8%. An explanation could come from contextual information from the scene that could prevent some confusion. The chessboard images that hardly benefit from contextual information at all and contain real repetition jump with respectively 36.99% and 50.46%.

4 CONCLUSIONS

We presented the CORE algorithm, a pre-processing filter which extract from a feature vector set a smaller subset less subject to confusion by removing highly

Table 1: Comparisons of the results (percentage, number of good matches/total matches) for three different approaches: first column plain matching SIFT, second column SIFT with the 2NN filter ($d = 0.8$) and last column SIFT with both CORE ($p = 0.1$) and 2NN filter ($d = 0.8$).

couple	unfiltered		2NN		CORE + 2NN	
object0014	23.89%	*322 / 1348*	70.68%	*258 / 365*	81.82%	*153 / 187*
object0008	20.00%	*336 / 1680*	52.71%	*204 / 387*	66.51%	*143 / 215*
object0039	26.78%	*448 / 1673*	66.24%	*310 / 468*	67.37%	*159 / 236*
object0110	24.58%	*222 / 903*	57.29%	*165 / 288*	69.34%	*95 / 137*
object0164	25.16%	*685 / 2723*	65.66%	*545 / 830*	71.88%	*317 / 441*
object0170	41.61%	*928 / 2230*	80.25%	*760 / 947*	87.83%	*469 / 534*
object0181	32.35%	*645 / 1994*	74.77%	*495 / 662*	81.69%	*290 / 355*
object0192	18.75%	*486 / 2592*	64.78%	*309 / 477*	73.93%	*241 / 326*
object0106	25.06%	*505 / 2015*	74.71%	*325 / 435*	77.42%	*216 / 279*
chess01	15.92%	*225 / 1413*	47.49%	*142 / 299*	84.48%	*49 / 58*
chess02	10.72%	*182 / 1698*	35.98%	*127 / 353*	86.44%	*51 / 59*

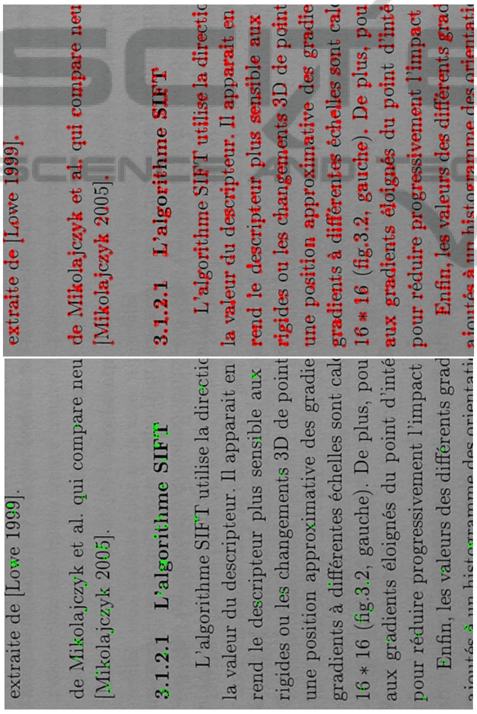

Figure 4: Example from our personnal image document dataset, top is keypoints removed and bottom is keypoints kept. $p = 0.1$.

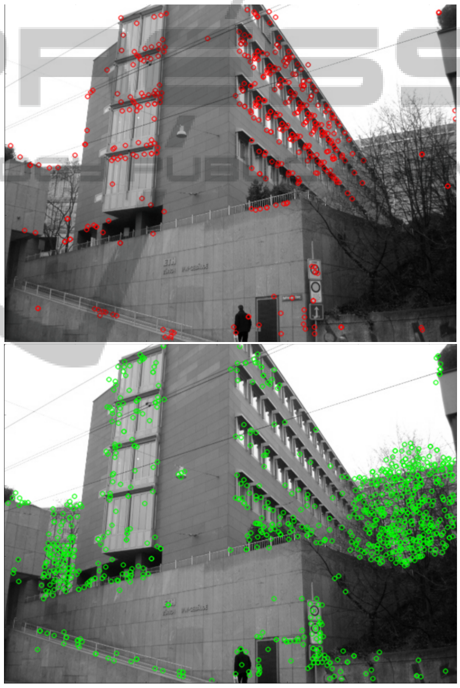

Figure 5: Example from Zurich dataset, top is keypoints removed and bottom is keypoints kept. $p = 0.1$.

similar keypoints thanks to a probability approach. Results showed that subsets extracted are more discriminant and our approach can be combined with post-processing ones.

However, due to the kernel density estimator used, our algorithm can only be applied on floating point based descriptors, putting aside the recent developments in the binary descriptors field. A binary version of CORE will require a very different approach and this will be the subject of future work.

ACKNOWLEDGEMENTS

This work was financially supported by the French region Provence-Alpes-Côte d'Azur (PACA).

REFERENCES

Bay, H., Tuytelaars, T., and Gool, L. (2006). Surf: Speeded up robust features. In *Computer Vision ECCV 2006*, volume 3951 of *Lecture Notes in Computer Science*, pages 404–417. Springer Berlin Heidelberg.

Belongie, S., Malik, J., and Puzicha, J. (2002). Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522.

Calonder, M., Lepetit, V., Strecha, C., and Fua, P. (2010). Brief: Binary robust independent elementary features. In *Computer Vision ECCV 2010*, volume 6314 of *Lecture Notes in Computer Science*, pages 778–792. Springer Berlin Heidelberg.

Deriche, R., Zhang, Z., Luong, Q.-T., and Faugeras, O. (1994). Robust recovery of the epipolar geometry for an uncalibrated stereo rig. In *Proceedings of the Third European Conference on Computer Vision (Vol. 1)*, ECCV '94, pages 567–576, Secaucus, NJ, USA. Springer-Verlag New York, Inc.

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395.

Harris, C. and Stephens, M. (1988). A combined corner and edge detector. In *In Proc. of Fourth Alvey Vision Conference*, pages 147–151.

Jiang, N., Tan, P., and Cheong, L. F. (2011). Multi-view repetitive structure detection. In *ICCV*, pages 535–542. IEEE.

Lee, S., Collins, R., and Liu, Y. (2008). Rotation symmetry group detection via frequency analysis of frieze-expansions. In *Proceedings of CVPR 2008*.

Leutenegger, S., Chli, M., and Siegwart, R. Y. (2011). Brisk: Binary robust invariant scalable keypoints. In *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, pages 2548–2555, Washington, DC, USA. IEEE Computer Society.

Li, X. and Hu, Z. (2010). Rejecting mismatches by correspondence function. *Int. J. Comput. Vision*, 89(1):1–17.

Liu, Y., Collins, R., and Tsin, Y. (2004). A computational model for periodic pattern perception based on frieze and wallpaper groups. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(3):354–371.

Lowe, D. (1999). Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157 vol.2.

Loy, G. and Eklundh, J.-O. (2006). Detecting symmetry and symmetric constellations of features. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part II*, ECCV'06, pages 508–521, Berlin, Heidelberg. Springer-Verlag.

Ma, J., Zhao, J., Tian, J., Yuille, A. L., and Tu, Z. (2014). Robust point matching via vector field consensus. *IEEE Transactions on Image Processing*, 23(4):1706–1721.

Martins, P., Carvalho, P., and Gatta, C. (2012). Context aware keypoint extraction for robust image representation. In *Proceedings of the British Machine Vision Conference*, pages 100.1–100.12. BMVA Press.

Mikolajczyk, K. and Schmid, C. (2005). A performance evaluation of local descriptors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(10):1615–1630.

Mok, S. J., Jung, K., Ko, D. W., Lee, S. H., and Choi, B.-U. (2011). Serp: Surf enhancer for repeated pattern. In *Proceedings of the 7th International Conference on Advances in Visual Computing - Volume Part II*, ISVC'11, pages 578–587, Berlin, Heidelberg. Springer-Verlag.

Morel, J.-M. and Yu, G. (2009). Asift: A new framework for fully affine invariant image comparison. *SIAM J. Img. Sci.*, 2(2):438–469.

Mortensen, E. N., Deng, H., and Shapiro, L. (2005). A sift descriptor with global context. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 184–190, Washington, DC, USA. IEEE Computer Society.

Ortiz, R. (2012). Freak: Fast retina keypoint. In *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR '12, pages 510–517, Washington, DC, USA. IEEE Computer Society.

Pang, S., Xue, J., Tian, Q., and Zheng, N. (2014). Exploiting local linear geometric structure for identifying correct matches. *Computer Vision and Image Understanding*, 128(0):51 – 64.

Parzen, E. (1962). On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):pp. 1065–1076.

Pauly, M., Mitra, N. J., Wallner, J., Pottmann, H., and Guibas, L. J. (2008). Discovering structural regularity in 3d geometry. In *ACM SIGGRAPH 2008 Papers*, SIGGRAPH '08, pages 43:1–43:11, New York, NY, USA. ACM.

Rabin, J., Gousseau, Y., and Delon, J. (2007). A contrario matching of local descriptors.

Rosten, E. and Drummond, T. (2006). Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, volume 1, pages 430–443.

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, pages 2564–2571, Washington, DC, USA. IEEE Computer Society.

Sattler, T., Leibe, B., and Kobbelt, L. (2009). Scramsac: Improving ransac's efficiency with a spatial consistency filter. In *ICCV*, pages 2090–2097. IEEE.

Shao, T. S. H. and Gool, L. V. (2003). Zubud-zurich buildings database for image based recognition, technical report no. 260.

T. Trzcinski, M. C. and Lepetit, V. (2013). Learning Image Descriptors with Boosting. *submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

Torr, P. H. S. and Murray, D. W. (1995). Outlier detection and motion segmentation. pages 432–443.

Torr, P. H. S. and Zisserman, A. (2000). Mlesac: A new robust estimator with application to estimating image geometry. *Comput. Vis. Image Underst.*, 78(1):138–156.

Trzcinski, T. and Lepetit, V. (2012). Efficient Discriminative Projections for Compact Binary Descriptors. In *European Conference on Computer Vision*.

Wu, C. (2007). SiftGPU: A GPU implementation of scale invariant feature transform (SIFT). http://cs.unc.edu/ ccwu/siftgpu.

Zhao, J., Ma, J., Tian, J., Ma, J., and Zhang, D. (2011). A robust method for vector field learning with application to mismatch removing. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2977–2984.