Beyond Bits: Reconstructing Images from Local Binary Descriptors

Emmanuel d'Angelo, Alexandre Alahi and Pierre Vandergheynst
Swiss Federal Institute of Technology
emmanuel.dangelo@epfl.ch

Abstract

Local Binary Descriptors (LBDs) are good at matching image parts, but how much information is actually carried? Surprisingly, this question is usually ignored and replaced by a comparison of matching performances. In this paper, we directly address it by trying to reconstruct plausible images from different LBDs such as BRIEF [4] and FREAK [1]. Using an inverse problem framework, we show that this task is achievable with only the information in the descriptors, excluding the need of additional data. Hence, our results represent a novel justification for the performance of LBDs. Furthermore, since plausible images can be inferred using only these simple measurements, this emphasizes the concerns about privacy and secrecy of image keypoints raised by [12], that could have an important impact on public applications of image matching.

1. Introduction and background

What is a good keypoint ? This question is central for many image-based Pattern Recognition and Computer Vision algorithms, that rely on detecting and matching interest points. The answer is commonly split into two independent qualities: good detection and good description. The detection quality is usually assessed following the work of Mikolajczyk and Schmid [9] by measuring the stability (under geometric changes) and the accuracy (with respect to a pose estimation task) of the detector. On the other hand, descriptors are judged from their performance in retrieval challenges [11], implicitly evaluating their descriptive and discriminative power.

We propose instead to explore an explicit approach to measure the amount of local content captured by a descriptor by attempting to reconstruct the original geometry. We focus on the increasingly popular Local Binary Descriptors (LBDs), mainly through two instances: the Binary Robust Independent Elementary Features (BRIEF) [4] and the Fast Retina Keypoint (FREAK) [1].

LBDs encode the values of pixel-wise or patch-wise differences in the region of interest, providing a *spatialized* information. Note that it is still possible to address the reconstruction task from descriptors based on integral measures, such as SIFT [8] and SURF [3], as shown by [12], but it requires some additional information such as an a priori database of patches indexed by their descriptors [12]. In contrast, our goal is to see how far one can go *without* any external information.

1.1. Local Binary Descriptors

Given an image patch p, forming an LBD of size d involves two steps: computation of a feature vector $\mathcal{L}(p) \in \mathbb{R}^d$ followed by its binarization in $\{0,1\}^d$.

Each component of $\mathcal{L}(p)$ is computed by:

$$\mathcal{L}(p)_i = \langle \mathcal{G}_{x_i,\sigma_i}, p \rangle - \langle G_{x'_i,\sigma'_i}, p \rangle, \qquad (1)$$

where $\mathcal{G}_{x,\sigma}$ denotes a Gaussian of width σ centered in x, and $\langle \cdot, \cdot \rangle$ is the usual scalar product (see Fig. 1). The quantization step is a simple thresholding: $\mathcal{Q}(p)_i = 1$ if $\mathcal{L}(p)_i$ is strictly positive, and 0 otherwise.

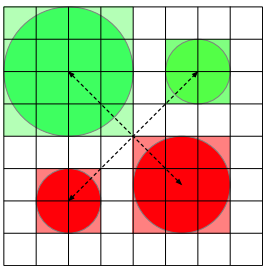

Figure 1. Computation of $\mathcal{L}(p)$ in the case $d = 2$. The Gaussian mean of each red area is subtracted from the mean of the corresponding green area.

Typically, a 32-by-32 pixels image patch (1024 values) is reduced to $d = 512$ differences, i.e. only 64 bytes

Preprint submitted to 21st International Conference on Pattern Recognition.
Received March 22, 2012.

after quantization. Hence, LBDs are very compact. Furthermore, they are computationally very efficient: only difference and thresholding operations are involved, and matching scores are computed by fast XOR operations to form Hamming distances. Added to their state-of-the-art results in recognition tasks [1], these qualities make them suitable for mobile applications where hardware resources are a limiting factor.

1.2. The BRIEF and FREAK descriptors

LBDs differ by the way they pick up the measurement $\mathcal{G}_{x_i,\sigma_i}$. The pioneering BRIEF uses randomly selected positions x_i with a fixed (small) radius σ_i. On the other hand, FREAK relies on a retinal sampling pattern with increasing radii where the most relevant Gaussians are kept according to a greedy selection process. As a consequence, outer points have a low spatial resolution, while the center of a patch is finely described.

2. Reconstruction as an inverse problem

In the sequel, we focus on the reconstruction from the non-quantized vector $\mathcal{L}(p)$. Although there is no convergence guarantee, our experiments in Sec. 3 show that the proposed algorithm can also deal with binarized descriptors $\mathcal{Q}(p)$.

2.1. Inverse problem for a single patch

Stacking all its values in column, we identify the patch p with the corresponding vector of $\mathbb{R}^{n^2}$. The averaging-then-differentiating process of Eq. (1) can then be expressed as a row-column product. Considering all the d components of $\mathcal{L}(p)$ together, their computation can be expressed as a matrix-vector product:

$$\mathcal{L}(p) = A_{\mathcal{L}} p, \tag{2}$$

where $A_{\mathcal{L}}$ is a sparse matrix with d rows and n^2 columns whose non-zero coefficients correspond to the colored pixels in Fig. 1.

The reconstruction problem of a patch from its description is obviously ill-posed, since $d \ll n^2$. Classically, we tackle this difficulty by using additional constraints: the resulting patch $\hat{p}$ should have a small Total Variation (TV) and belong to an acceptable set $\mathcal{S}$ that will be made precise later.

Given a descriptor $g = \mathcal{L}(p)$, the reconstruction process can now be cast as a *regularized inverse problem*:

$$\hat{p} = \underset{p}{\operatorname{argmin}} \underbrace{\lambda \|A_{\mathcal{L}} p - g\|_1}_{\text{data term}} + \underbrace{\|p\|_{\text{TV}} + \delta_{\mathcal{S}}(p)}_{\text{regularization}}, \tag{3}$$

where $\|\cdot\|_{\text{TV}} = \sqrt{\|\nabla p\|_2^2}$ is the TV semi-norm, and $\delta_{\mathcal{S}}(\cdot)$ is the indicator function of $\mathcal{S}$ defined by:

$$\delta_{\mathcal{S}}(p) = \begin{cases} 0 & \text{if } p \in \mathcal{S} \\ +\infty & \text{otherwise.} \end{cases} \tag{4}$$

Eq. (3) states that an estimated patch $\hat{p}$ should have a descriptor close to the observation g (in the ℓ_1 sense), while being piecewise smooth and belonging to $\mathcal{S}$. We have chosen the ℓ_1-norm in the data term for its robustness against actual matching error values, which makes it a good candidate to address the task of reconstruction from binarized measurements.

Since only differences are stored, the output of LBDs is $\mathbf{0}$ for any constant input. Hence, patches can be reconstructed only up to a constant. Assuming pixel values in the interval $[0, 1]$, we arbitrarily fix the mean value of the patches to 0.5 and define $\mathcal{S}$ by:

$$\mathcal{S} = \{p \in \mathbb{R}^{n^2} \text{ s.t. } \bar{p} = 0.5 \text{ and } \|p\|_\infty \leqslant 1\}, \tag{5}$$

where $\bar{p}$ is a shortcut for the mean ($\bar{p} = \frac{1}{n^2} \sum_{i=1}^{n^2} p_i$).

2.2. Primal-dual optimization background

To solve Eq. (3), we use the primal-dual algorithm introduced in [6] and recall here the main results for completeness.

We consider a generic primal problem of the form:

$$\hat{x} = \underset{x}{\operatorname{argmin}}\ F(Kx) + G(x), \tag{6}$$

where K is a linear operator, and F and G convex (non-smooth) functions. Introducing F^*, the Fenchel-Legendre transform of F, Eq. (6) can be recast as a primal-dual problem on F^*, G, x and its dual variable y. Let us also define the *proximal mapping* $\text{prox}_\sigma[F]$ of a function F with:

$$\text{prox}_\sigma[F](x) = \underset{z}{\operatorname{argmin}}\ F(z) + \frac{\|x - z\|_2^2}{2\sigma}. \tag{7}$$

Then, given an upper bound L on the norm of K and the expressions of $\text{prox}_\sigma[F^*]$ and $\text{prox}_\tau[G]$, Alg. 1 provides the solution of Eq. (6) (see [6] for a proof):

Algorithm 1 Basic Chambolle-Pock algorithm.

1: Take $L \geqslant \|K\|^2$, choose τ, σ, θ such that $L^2 \sigma\tau \leqslant 1$ and $\theta \in [0, 1]$
2: Initialize: $x_0 \leftarrow 0$, $y_0 \leftarrow 0$, $\bar{x}_0 \leftarrow x_0$
3: **for** $i = 0$ **to** $n - 1$ **do**
4: $\quad y_{i+1} \leftarrow \text{prox}_\sigma[F^*](y_i + \sigma K \bar{x}_i)$
5: $\quad x_{i+1} \leftarrow \text{prox}_\tau[G](x_i - \tau K^T y_{i+1})$
6: $\quad \bar{x}_{i+1} \leftarrow x_{i+1} + \theta(x_{i+1} - x_i)$
7: **end for**
8: **return** x_n

While it may seem inefficient to solve both primal and dual problems, this becomes interesting when the constraints on x can be computed through simple proximal mappings on either the primal or dual variable. For example, TV can be minimized through a robust pointwise thresholding of the dual [5, 6].

2.3. Patch reconstruction algorithm

From Eq. (2), it is clear that $A_{\mathcal{L}}$ is a linear operator. Then, choosing $K = \begin{pmatrix} A_{\mathcal{L}} \\ \nabla \end{pmatrix}$, $y = A_{\mathcal{L}}p$ and $z = \nabla p$ transforms Eq. (3) to the desired form of Eq. (6) with:

$$\begin{aligned} F(Kp) &= F_1(y) + F_2(z) \\ &= \lambda\|y - g\|_1 + \|z\|_1, \qquad (8) \\ G(p) &= \delta_{\mathcal{S}}(p). \qquad (9) \end{aligned}$$

Dualizing with respect to F, we introduce the dual variables q and r of y and z respectively. The corresponding conjugate functions $F_1^*(q)$ and $F_2^*(r)$ are:

$$F_1^*(q) = \delta_{Box(\lambda)}(q) + \langle q, g\rangle_G, \qquad (10)$$
$$F_2^*(r) = \delta_{Box(1)}(|r|), \qquad (11)$$

where $\delta_{Box(\lambda)}$ is the ℓ_∞-ball of radius λ:

$$\delta_{Box(\lambda)}(x) = \begin{cases} 0 & \text{if } \|x\|_\infty \leqslant \lambda \\ \infty & \text{otherwise.} \end{cases} \qquad (12)$$

Using $\nabla^T = -\operatorname{div}$, it is straightforward to see that $K^T\begin{pmatrix} q & r \end{pmatrix} = A_{\mathcal{L}}^T q - \operatorname{div} r$ in line 5 of Alg. 1.

Finally, one needs the different proximal mappings. Since G is an indicator function, $\operatorname{prox}_\tau[G]$ is obtained by projection onto $\mathcal{S}$, which is a hyperplane of equation $(1/n^2)\sum_{i=1}^{n^2} p_i = 0.5$, and the components of the result are clipped to $[0, 1]$. The proximal mappings of $F_1^\star$ and F_2^* can be obtained through variational calculus:

$$\operatorname{prox}_\sigma[F_1^\star](q) = \operatorname{sign}(q - \sigma g) \cdot \max(\lambda, |q - \sigma g|), \qquad (13)$$

$$\operatorname{prox}_\sigma[F_2^\star](r) = \frac{r}{\max(\mathbf{1}, |r|)}, \qquad (14)$$

where all operations are taken *component-wise* and *pixel-wise*.

3. Results and discussion

3.1. Implementation

LBDs are indeed local descriptors. Hence, to reconstruct images, we first work on a subset of patches. Then, we merge the reconstructed patches by averaging on their overlap to form the final image.

To check the consistency of the reconstruction quality with retrieval performances, we have tested three different operators: BRIEF [4], FREAK [1], and a randomized selection of the pairs used for the training of FREAK. All the operators were reimplemented in C++ using the same codebase for a fair comparison. Algorithmic parameters were kept identical.

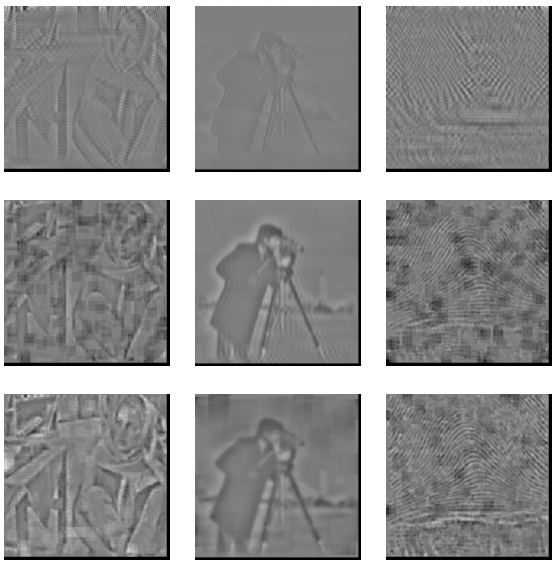

Figure 2. Reconstruction from floating-point descriptors $\mathcal{L}(p)$ of 512 pairs. From top to bottom: BRIEF, randomized FREAK, FREAK.

3.2. Reconstruction results

Fig. 2 shows some examples of reconstructing different classical images, for various patch sizes and overlap. In these images, one can clearly recognize the depicted objects. It is remarkable that the reconstruction results almost look like Laplacian filtered versions of the images, emphasizing the edges.

The hierarchy of the patterns is respected: the optimized FREAK behaves better than the randomized version, and better than BRIEF. This ranking is emphasized when the patch size grows: `barbara` was reconstructed using 32-by-32 patches, and `cameraman` using 64-by-64 patches.

Fig. 3 shows some preliminary results obtained by applying the proposed algorithm on binarized FREAK descriptors $\mathcal{Q}(p)$. No modification is made to the implementation. In particular, the operator used in the recon-

Preprint submitted to 21st International Conference on Pattern Recognition.
Received March 22, 2012.

struction process does not include the binarization step and remains linear. Yet, these first results are clearly promising.

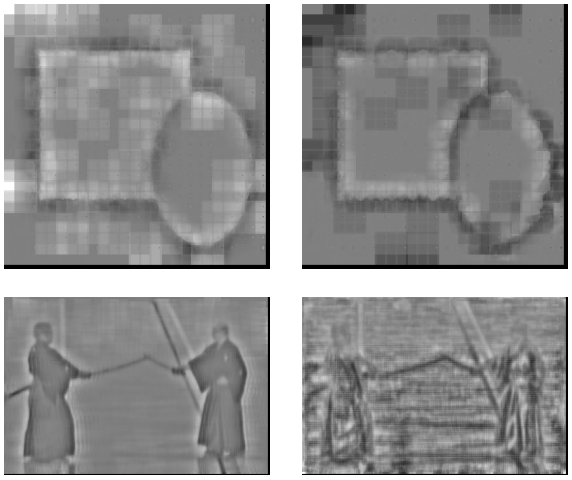

Figure 3. Preliminary results from binarized descriptors. Left: using floating-point FREAK. Right: binarized FREAK.

Finally, Fig. 4 shows the result of reconstructing only patches that were selected by the FAST detector [10]. Interestingly, the image content is near recognizable, hence also justifying a posteriori the definition of FAST.

Figure 4. Left: original kata image. Right: reconstruction of FREAK descriptors computed around each FAST keypoint.

4. Conclusion and future work

In this work, we have shown that it is actually possible to reconstruct image parts from their local descriptors without any additional information, extending previous results from [12]. Furthermore, the proposed algorithm is robust enough to handle, at least partially, binarized descriptors, that carry very few bits of information.

Our results provide another perspective on the performance analysis of LBDs: descriptors that lead to the best reconstructions also lead to the best retrieval results. Application developers will be interested in the privacy and secrecy issues raised here and in [12]. We plan to improve the proposed algorithm to reliably handle quantized descriptors by integrating techniques from 1-bit Compressive Sensing [7].

Acknowledgments The authors would like to thank Dr. Laurent Jacques for his insightful suggestions. The `kata` picture is courtesy of Flickr user `Sansuiso` and retrieved from the dataset [2].

References

[1] A. Alahi, R. Ortiz, and P. Vandergheynst. FREAK: Fast Retina Keypoint. In *IEEE Conference on Computer Vision and Pattern Recognition (To Appear)*, 2012.

[2] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen. Interactively Co-segmentating Topically Related Images with Intelligent Scribble Guidance. *International Journal of Computer Vision*, 93(3):273–292, Jan. 2011.

[3] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. *Computer Vision–ECCV 2006*, pages 404–417, 2006.

[4] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. BRIEF: Binary Robust Independent Elementary Features. *Computer Vision–ECCV 2010*, pages 778–792, 2010.

[5] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1):89–97, 2004.

[6] A. Chambolle and T. Pock. A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, Dec. 2010.

[7] L. Jacques, J. N. Laska, P. T. Boufounos, and R. G. Baraniuk. Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors. *arXiv.org*, 2011.

[8] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

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

[10] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. *Computer Vision–ECCV 2006*, pages 430–443, 2006.

[11] P. Turcot and D. Lowe. Better matching with fewer features: The selection of useful features in large database recognition problems. In *Computer Vision Workshops (IEEE ICCV Workshops), 2009*, pages 2109–2116, 2009.

[12] P. Weinzaepfel, H. Jegou, and P. Pérez. Reconstructing an image from its local descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 337–344, 2011.