DOI: 10.2352/ISSN.2470-1173.2016.14.IPMVA-387

STABLE: Stochastic Binary Local Descriptor for High-performance Dense Stereo Matching

Svorad Štolc, Kristián Valentín, Reinhold Huber-Mörk; AIT Austrian Institute of Technology GmbH, Intelligent Vision Systems, Digital Safety & Security Department, Donau-City-Straße 1, 1220 Vienna, Austria

Abstract

We propose a novel stochastic binary local descriptor (STABLE) specifically designed for dense stereo matching in high-performance vision applications. STABLE is a local binary descriptor which builds upon the principles of the compressed sensing theory. The most important properties of STABLE are the independence of the descriptor length from the matching window size and the possibility that more than one pair of pixels contributes to a single descriptor bit. Individual descriptor bits are computed by comparing image intensities over pairs of balanced random sub-sets of pixels chosen from the whole described area. On a synthetic as well as real-world examples we demonstrate that STABLE provides competitive or superior performance than other state-of-the-art local binary descriptors in the task of dense stereo matching. We show that STABLE performs significantly better than the census transform (CT) and local binary patterns (LBP) in all considered geometric and radiometric distortion categories to be expected in practical applications of stereo vision. Moreover, we show as well that STABLE provides comparable or better matching quality than the binary robust independent elementary features (BRIEF) descriptor. The low computational complexity and flexible memory footprint makes STABLE well suited for most hardware architectures.

Introduction

Feature point detection and description is used in various applications of computer vision. Feature point detectors typically localize points of high saliency sparsely distributed over images. Feature point description encodes the local vicinity of a feature point, or any pixel in general, into a numerical representation which aims to fulfill several goals such as being highly discriminative, precisely localized and invariant w.r.t. radiometric and geometric distortions. Dense matching assigns such a descriptor to each image pixel. This makes the goal of being efficient with respect to memory (i.e. low descriptor length), and speed (i.e. time for computing and matching the descriptors), more important. On the other hand, for sparsely sampled key points and more general applications, more resource demanding descriptors are affordable, such as SIFT [1] which is by default a floating point vector with 128 elements.

Since the introduction of SIFT a number of feature detectors and descriptors were suggested over the last decades [1]. Among others, the goal of speeding up SIFT was met in SURF [2]. Some representations of local derivatives, e.g. gradient orientation histograms, are commonly used in those descriptors. Higher speed is sometimes also traded against reduced invariance properties, e.g. in BRIEF [3]. Efficient representations and fast matching is obtained by the family of binary descriptors. The ORB is an alternative to SIFT and SURF being based on a binary description [4].

In this paper, we introduce the *stochastic binary local descriptor* (STABLE). It belongs to a broad class of local binary descriptors, along with the census transform (CT) [5], local binary patterns (LBP) [6], binary robust independent elementary features (BRIEF) [3], binary robust invariant scalable keypoints (BRISK) [7] or fast retina keypoints (FREAK) [8]. The most similar descriptor to STABLE is BRIEF [3], where the main difference lies in the ability of STABLE to have more than one pair of pixels contributing to a single descriptor bit.

STABLE can be related to the principle of *compressed sampling* [9]. The compressed sampling theory claims that each signal with a sparse representation in some (potentially unknown) linear basis can be preserved and reconstructed from a small number of random projections. For natural images this means that, due to the sparsity of image edges and inherent smoothness, it is sufficient to sample the image in a compressive manner without loosing any significant information. While the reconstruction is not the main focus in our application, we exploit the principles of compressed sampling just for deriving an efficient binary representation of any given pattern, i.e. for encoding the pattern into a constant number of bits that is greatly independent from the pattern's size.

The paper is organized as follows. We start with reviewing stereo matching descriptors and their evaluation, especially local binary descriptors. Then we describe STABLE in detail and provide results of the ROC analysis of STABLE in comparison with CT, LBP and BRIEF. A real-world stereo vision example provides a visual comparison of the matchin quality obtained by STABLE and CT. Finally, we end with conclusions.

Descriptors for stereo matching

In stereo imaging the range for each pixel is obtained from the estimated disparity, i.e. the displacement between corresponding points observed in two (or more) images. The epipolar constraint in stereo vision states that a point in one image is found along the corresponding epipolar line in the other image. Epipolar rectification of stereo image pairs aligns epipolar lines to images lines, thus reducing the correspondence estimation to a search oriented along an expected disparity range in image lines. Corresponding points are typically identified via block matching, i.e. comparison of image patches. Measures of block similarity include direct comparison of pixel intensities using similarity metrics such as the sum of absolute differences (SAD), the sum of squared errors (SSE), the normalized cross-correlation (NCC), and comparison based on the descriptors mentioned above. While

DOI: 10.2352/ISSN.2470-1173.2016.14.IPMVA-387

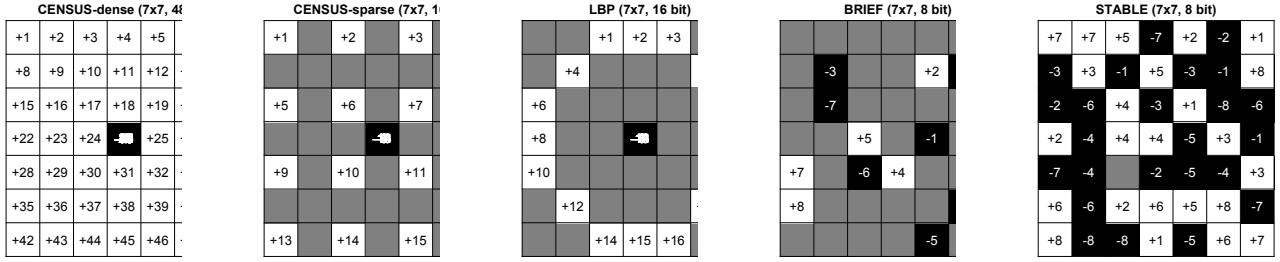

Figure 1: Examples of index filter masks of different binary feature descriptors defined on the 7x7 pixel matching window.

for descriptors such as SURF or SIFT some vector metrics in high-dimensional spaces are commonly used to quantify descriptor similarity, for binary descriptors the Hamming distance is typically applied.

Evaluation of detectors and descriptors

Scale and angle invariance has been a major topic in the study of local interest detectors so far, e.g. repeatability characterizes the rate at which a set of corresponding tie points is detected in sets of images [10]. Alternatively, repeatability and matching score are used to evaluate correct matches [11]. On the other hand, in order to characterize local descriptors measure to quantize reliable, stable and precise localization are of major interest. Evaluation of descriptors was performed using recall and precision under several geometric and radiometric distortions [12]. Beyond this, we are also interested in stability of the descriptor depending on the compactness of its representation, i.e. the number of descriptor bits used.

Local binary descriptors

In general, binary descriptors have been used for tasks like texture analysis, recognition and matching, e.g. local binary patterns (LBP) [6], [13] and the census transform (CT) [5]. In the context of local descriptors several fast binary descriptors were also developed recently, e.g. BRIEF [3], BRISK [7], FREAK [8], and some more. In our experiments we considered the *center-based* descriptors CENSUS and LBP, where center-based refers to the fact that pairwise comparison always involve the central pixel, and the *uncentered* descriptors BRIEF and STABLE. The main difference in binary descriptors is in the sampling pattern for local intensity comparisons which results in a binary descriptor vector. The CENSUS-dense descriptor is the only descriptor utilizing exactly all pixels in the considered matching window. We alternatively investigate the CENSUS-sparse descriptor which uses a sub-sample of off-center pixels on a regular grid and compare those against the central pixel. The BRIEF descriptor uses a sub-sample of pixel pairs (typically sparse) located at arbitrary positions in the matching window. The resulting descriptor lengths equals the number of pixel pair comparisons performed. Finally, with STABLE we also get pixel pairs at random positions, but we are able to map a larger number of pixel pairs to a smaller number of descriptor bits. Fig. 1 summarizes the compared descriptor masks.

The STABLE descriptor

We consider an image patch $\mathbf{p}$ of size $X \times Y$ pixels. The operation β derives the i-th descriptor bit $d_i \in \mathbf{d}$ from patch $\mathbf{p}$ as follows:

$$\beta(\mathbf{p},i) = \begin{cases} 1 & \text{if } (\mathbf{p} * f_i) > 0, \\ 0 & \text{otherwise.}, \end{cases} \tag{1}$$

where f_i is a filter mask of equal size as the image patch $\mathbf{p}$. We refer to the operation β as the *binarized convolution*. The filter dictionary $\mathbf{f}$ contains K sparse filter masks f_i. Each entry in f_i is either 0, 1 or -1. The descriptor $\mathbf{d}$ is a K-dimensional bitmask which is obtained for a given image patch $\mathbf{p}$ using

$$\mathbf{d}(\mathbf{p}) = \sum_{i=1}^{K} 2^{i-1} \beta(\mathbf{p},i). \tag{2}$$

Fig. 2 shows this operation schematically, a set of sparse filter masks from a dictionary are applied to the same image patch and, depending on the number and individual signs of the filter mask entries, a number of pixels is contributing to each descriptor bit.

A more efficient implementation of STABLE, avoiding binarized convolution with K sparse feature filters, uses a single index filter mask $\mathbf{g}$. This mask $\mathbf{g}$ is of the same size as the image patch $\mathbf{p}$ and encodes at non-zero pixel positions the position

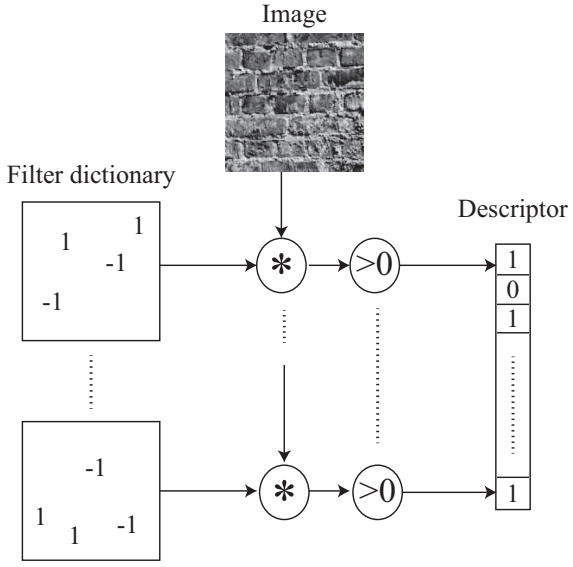

Figure 2: Operation of the STABLE descriptor: Sparse filters form a dictionary where each filter mask mostly consists of entries of 0, other entries $\{-1,1\}$ are randomly distributed. An image patch is convolved with each filter mask and the result is thresholded (*binarized convolution*) and inserted into descriptor bits.

DOI: 10.2352/ISSN.2470-1173.2016.14.IPMVA-387

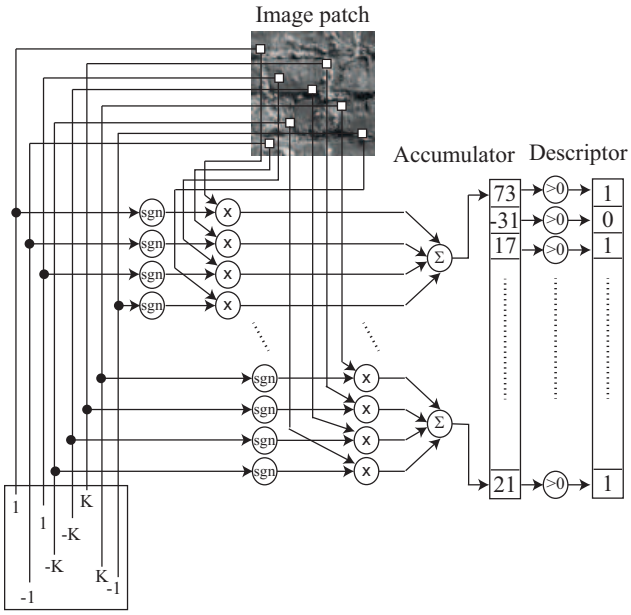

Figure 3: Efficient implementation of the STABLE descriptor: An index filter mask contains pixel indices and signs. An image patch is accessed using this mask and a signed sum is inserted into an accumulator array. The descriptor is finally obtained by binarization of the accumulator entries.

in the descriptor array **d** and a sign. An accumulator array a of size K is used to perform a sign-dependent accumulation in cell i of the pixel values in **p** with corresponding filter mask index $|i|, i = 1, \ldots, K$. After all accumulators cells are processed the descriptor **d** is derived by thresholding each cell entry of a. The improved operation involving the filter index mask **g** instead of the filter dictionary **d** is shown in Fig. 3.

Computational complexity analysis

For computational complexity analysis, we compare STABLE and BRIEF with K features bits applied to $X \times Y$ image patch implemented using the index filter mask implementation which was shown in Fig. 3. In general, there are two main operations required for using any of the local binary descriptors – *building* and *matching*. The matching operation is typically identical for all binary descriptors, making use of the Hamming distance applied to binary strings of length K. The difference can thus be only in the computational complexity of the building operation.

Building of the descriptors is comprised of three basic steps:

(i) generating the index filter mask,
(ii) computing the accumulator values,
(iii) binarization of the accumulator values.

The index filter mask is generated only once and can be considered as an input parameter for the building operation. Therefore this step can be omitted from our analysis. The binarization step uses the same thresholding algorithm for both analyzed descriptors and can be neglected as well. Hence the only difference comes from the complexity of computing the accumulator values, as shown in Algorithm 1. While STABLE requires processing of $X \times Y$ elements from the index filter mask as well as from the image patch (or $X \times Y - 1$ for odd number of pixels), BRIEF requires to process only $2K$ such elements. Consequently, for a fixed K, STABLE scales linearly with the number of patch pixels while BRIEF, in principle, requires only a constant time.

Algorithm 1 Computation of the accumulator values in BRIEF and STABLE using a single index filter mask.

Require: image patch **p**, index filter mask **g**
  initialize array a to size K with values of 0
  for non-zero i in **g** **do**
    $a[|i|] \leftarrow a[|i|] + \mathrm{sgn}(i) \times \mathbf{p}[\text{position of } i \text{ in } \mathbf{g}]$
  end for

In practice, however, the difference between the actual execution time on CPU or GPU platforms and the theoretical one might be more in favor of STABLE due to caching in the on-chip memory. When a memory read for a cell is requested, often nearby cells are fetched and stored in the cache as well (details are hardware-dependent). To enable optimal caching, the data has to be well-organized in the memory (i.e. aligned with the hardware layout) and should be accessed using predictable memory access patterns (e.g. in the same order as they are stored). This is especially important for GPUs where the global memory latency is higher compared to the CPU memory and thus optimal utilization of the cache memory has higher impact of the final performance. We believe that such memory caching mechanisms can be better utilized with STABLE as all elements in both index as well as image patch arrays are always accessed and thus the memory access pattern can be fully optimized. On the other hand, as BRIEF uses a random-access sparse memory pattern, prediction algorithms implemented in various memory caching mechanisms are more prone to fail.

Experimental design

In order to evaluate performance of the STABLE descriptor compared with other state-of-the-art local binary descriptors, we employed a similar evaluation scheme as suggested in [12] based on the analysis of receiver operator characteristic (ROC) curves. We extracted 1200 grayscale patterns from 48 natural images contained in the data set introduced in [12], always 25 patterns per image at random locations. Given the perturbation type, for each pattern we introduced 25 synthetic perturbations which gave a total number of 30 000 patches. In this study we considered five different types of perturbations:

(i) Gaussian additive noise ($\sigma \leq -20\ dB$),
(ii) Gaussian blur ($\sigma \leq 4\ px$),
(iii) shift in random direction ($\leq 3\ px$),
(iv) scaling ($\leq \pm 10\ \%$),
(v) rotation ($\leq \pm 10\ deg$).

As for the representative matching window we considered patches of size 15×15 pixels.

Given the set of 30 000 patches defined for each perturbation type, there is always a group of 25 associated perturbed versions for each patch in the data set. Making every patch a query, one can assess its Hamming distance to all patches in the data set making use of a particular feature descriptor. Knowing that for each query there are only 25 relevant elements, one can calculate the precision and recall values for all result sets associated with different

DOI: 10.2352/ISSN.2470-1173.2016.14.IPMVA-387

thresholds put on the Hamming distance. The ROC curve is then defined by the obtained precision and recall values.

We compared in total five local binary descriptors:

(i) CENSUS-dense,
(ii) CENSUS-sparse,
(iii) LBP,
(iv) BRIEF,
(v) STABLE.

While for CENSUS and LBP the descriptor size depends on the matching window, in the case of STABLE and BRIEF the number of feature bits is defined independently from the matching window. Thus we also looked into the relationship between matching performance, expressed in terms of the area under the ROC curve (AUC), and the descriptor size in bits. Furthermore, as both these descriptors are generated stochastically, their performance was assessed as the average and standard deviation over 25 trials with different randomly generated filter masks. We believe that should provide a clear picture about the typical performance and stability of those stochastic descriptors.

Results

Fig. 4 shows the recognition performance obtained by different feature descriptors for a constant configuration of the descriptor size. Going from the worst to the best performing descriptors, it can be seen that the LBP provides the overall worst performance for all perturbation types. It is then followed by CENSUS-sparse and CENSUS-dense which both provide comparable performance despite their very different numbers of feature bits. For most perturbation types it is then followed by BRIEF and finally by STABLE (notice the curve with circles exceeds all the other curves in most cases).

In Fig. 5, the matching performance is analyzed in relationship with the descriptor size. All descriptors with a constant number of bits are marked as points, while all the others are represented as curves. In this analysis it is even more pronounced, that the performance of the both CENSUS descriptors as well as LBP is significantly worse than for STABLE and BRIEF at the respective bit counts. In the case of noise, blur, and shift perturbations, the STABLE descriptor outperforms the BRIEF descriptor, especially for medium numbers of feature bits.

The performance of STABLE vs. BRIEF is documented in detail in Fig. 6. The advantage of STABLE over BRIEF is expressed in terms of the recognition performance gain defined as a ratio between AUC values obtained by both descriptors using the same numbers of bits. It follows that AUC ratios above one mark the cases where STABLE outperformed BRIEF and vice versa. It is apparent that the advantage of STABLE is mostly pronounced for medium bit counts, while with the increasing size of the descriptor the difference is getting smaller as both descriptors become more similar to each other. It should be noted that at the maximum possible number of bits, both descriptors are in fact the same where each bit is generated by just a pair of pixels. There are two cases in which STABLE significantly outperformed BRIEF, namely perturbations by (i) the additive noise and (ii) the blur. In the case of additive noise, the performance gain as large as 30 % was obtained with 8 bit descriptors. For blur and shift perturbations, the highest AUC ratios exceeding 5 % were obtained for 32 bit and 8 bit descriptors, respectively. For scale and rotation perturbations, STABLE performs generally slightly worse than BRIEF, however the worst performance loss is still well below 5 %.

Real-world example

We present some results for an application of line-scan stereo acquisition and matching for asphalt pavement inspection. The purpose of this application is to assess 3D road surface as poor road conditions lead to increased wear and tear on vehicles and has as well an impact on surface water transport, noise emission, etc. Fig. 7 (a-b) shows a stereo image pair depicting a top down view onto a washed concrete surface. The estimated depth maps shown in Fig. 7 (c-d) are results of 15×15 CENSUS-dense and 15×15 STABLE with 64 bit descriptor length. The result of STABLE is less noisy (i.e. less "black" pixels) using just 64 bits, while achieving a qualitatively similar, or even slightly better, depth estimation as the $15 \times 15 - 1 = 224$ bit long CENSUS descriptor.

Fig. 8 shows the performance of STABLE with descriptor length ranging from 16 bits to 112 bits, which is the maximum bit count possible for the 15×15 matching window. While the 16 bit long descriptor still provides quite noisy results, using 32 or 64 bit descriptors improves the reconstruction quality significantly. On the other hand, increasing size of the descriptor to full 112 bits does not seem to improve the result any further.

Finally, Fig. 9 shows the influence of spatial averaging and additive noise on CENSUS-dense and STABLE. In both cases STABLE outperforms CENSUS-dense descriptor.

Conclusion

In this paper, we have introduced the STABLE descriptor suitable for high-performance dense stereo matching. STABLE is a novel stochastic binary local descriptor that relates to the compressed sensing theory for efficient representation of image patterns. We showed that STABLE provides significantly better matching quality w.r.t. the efficiency of data representation being preserved in a highly compressed binary form.

Compared with other state-of-the-art binary descriptors, our descriptor achieves the same matching quality with considerably fewer descriptor bits required, or alternatively, significantly better matching quality making use of the same number of descriptor bits. STABLE offers increased stability and robustness especially in the cases where data are subject to noise, blur, and/or slight misplacement, which is often observed in practice. Unlike some other descriptors the descriptor size and the matching window are defined independently in STABLE. Moreover, STABLE always utilizes all pixels of the given matching window for producing the required number of feature bits. That makes it suitable for many practical applications where a trade-off between the descriptor size, due to computational performance limitations, and the overall matching performance is necessary. Yet another indication of the same is that STABLE surpasses other analyzed descriptors predominantly in a small-medium range of feature bits.

Despite that STABLE requires more operations than BRIEF for the same input, their performance should be comparable for a reasonable matching window size, as STABLE can take better advantage of memory caching mechanisms implemented on different platforms including GPUs.

We have demonstrated that the proposed descriptor works very well for a broad class of natural patterns and that the in-

DOI: 10.2352/ISSN.2470-1173.2016.14.IPMVA-387

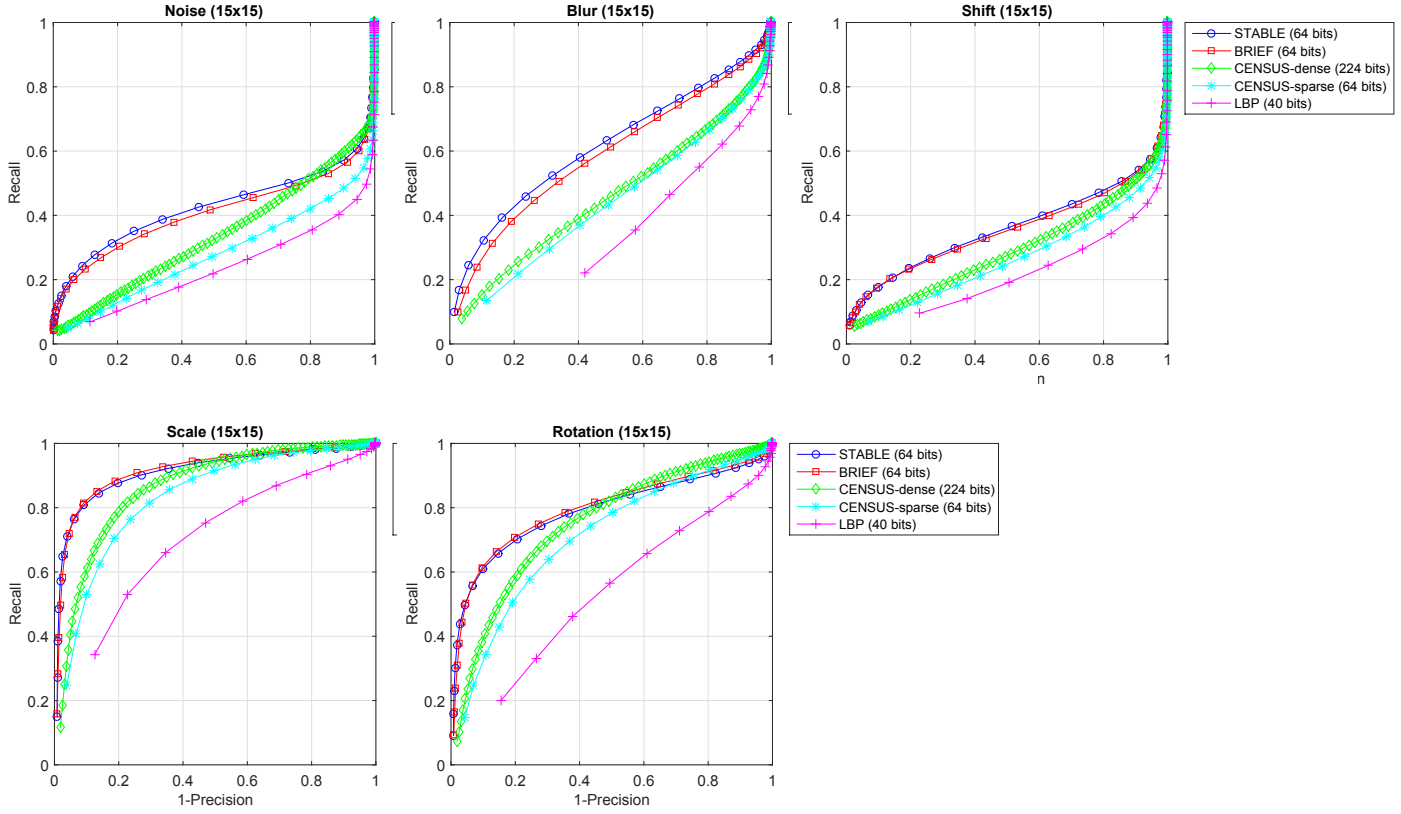

Figure 4: ROC curves obtained for different perturbation types. In the case of STABLE and BRIEF, the provided ROC curves represent the best performance case over 25 random trials, i.e. the one with the highest AUC value.

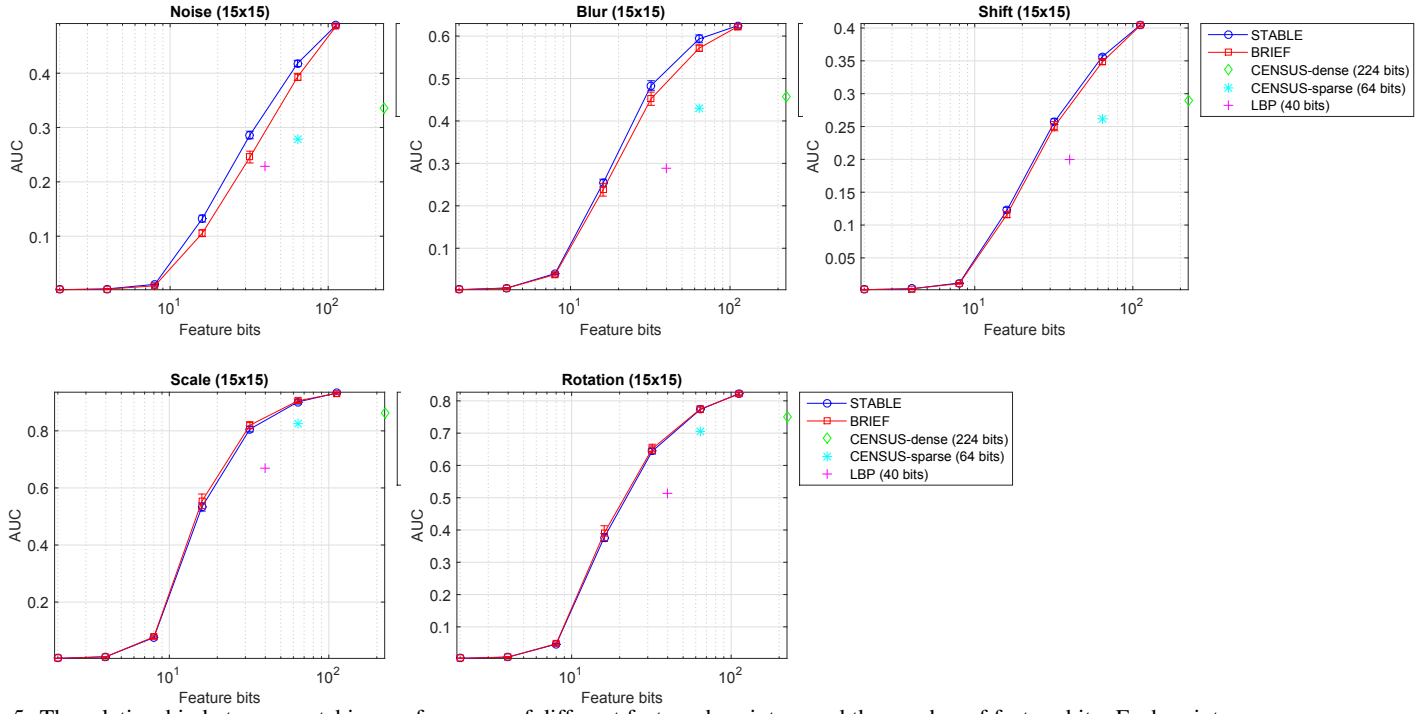

Figure 5: The relationship between matching performance of different feature descriptors and the number of feature bits. Each point on a curve stands for the average AUC value over 25 random trials, while the whiskers show the corresponding standard deviation.

DOI: 10.2352/ISSN.2470-1173.2016.14.IPMVA-387

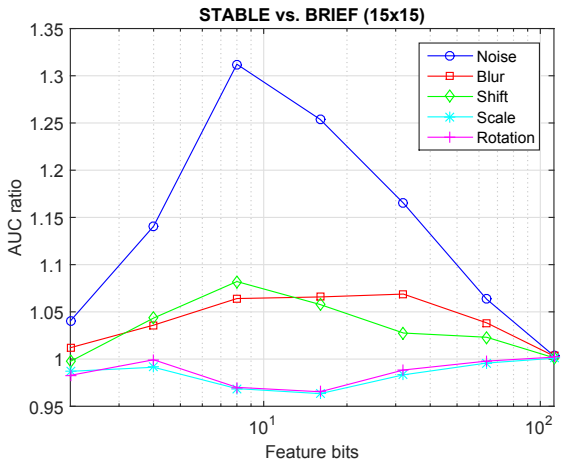

Figure 6: The recognition performance gain (i.e. the ratio between AUC values) of STABLE over BRIEF.

herent sparsity of those patterns suffice the assumptions of the compressed sensing theory. In the future we intend to look into the efficiency of STABLE when applied to more specific groups of non-natural patterns that arise from some special applications. Another direction of our future research will go towards ways of mitigating certain matching artifacts originating from a hard typically rectangular matching window, where each pixel is utilized precisely one time.

References

[1] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. of Comput. Vision*, 60(2):91–110, 2004.

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In Aleš Leonardis, Horst Bischof, and Axel Pinz, editors, *Proceedings of European Conference on Computer Vision (ECCV)*, volume 3951 of *Lecture Notes in Computer Science*, pages 404–417. Springer, Berlin Heidelberg, 2006.

[3] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *Proc. of European Conference on Computer Vision (ECCV)*, volume 6314 of *Lecture Notes in Computer Science*, pages 778–792. Springer Berlin Heidelberg, 2010.

[4] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An efficient alternative to SIFT or SURF. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 2564–2571, 2011.

[5] Ramin Zabih and John Woodfill. Non-parametric local transforms for computing visual correspondence. In Jan-Olof Eklundh, editor, *Proceedings of European Conference on Computer Vision (ECCV)*, volume 801 of *Lecture Notes in Computer Science*, pages 151–158, Stockholm, SE, 1994. Springer, Berlin Heidelberg.

[6] Mäenpää T. *The local binary pattern approach to texture analysis - extensions and applications*. PhD thesis, Machine Vision and Media Processing Unit, Infotech Oulu, University of Oulu, Finland, 2003.

[7] S. Leutenegger, M. Chli, and R.Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, pages 2548–2555, November 2011.

[8] A. Alahi, R. Ortiz, and P. Vandergheynst. Freak: Fast retina keypoint. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 510–517, June 2012.

[9] Richard Baraniuk. Compressive sensing. *IEEE Signal Processing Magazine*, 24(4):118–121, July 2007.

[10] C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of interest point detectors. *Int. J. of Computer Vision*, 37(2):151–127, 2000.

[11] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L.V. Gool. A comparison of affine region detectors. *Int. J. of Comp. Vision*, 65(1–2):43–72, 2005.

[12] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Trans. on Pat. Anal. Mach. Intell.*, 27(10):1615–1630, 2005.

[13] M. Pietikäinen, A. Hadid, G. Zhao, and T. Ahonen. *Computer Vision Using Local Binary Patterns*, volume 40 of *Computational Imaging and Vision*. Springer, London, 2011.

Author Biography

Svorad Štolc is a researcher at the Digital Safety & Security Department of AIT Austrian Institute of Technology GmbH, Vienna. In 2002, he earned his masters degree in Computer Science from Comenius University in Bratislava and, in 2009, PhD degree in Bionics and Biomechanics from Technical University of Košice and Slovak Academy of Sciences, Bratislava. His main research areas are image processing and computational imaging.

Kristián Valentín recieved his PhD in Computer Science from Comenius University in Bratislava, Slovakia in 2015. Since 2014, he works at AIT, Vienna, Austria in the field of computational imaging and computer vision.

Reinhold Huber-Mörk recieved his PhD in computer science from the University of Salzburg, Austria, in 1999. Since then he worked at the Aerosensing GmbH, Oberpfaffenhofen, Germany, in remote sensing image analysis, at the Advanced Computer Vision GmbH, Vienna, Austria, in computer vision and in 2006 he joined the AIT, Vienna, Austria, where he is currently senior scientist in the field of machine vision.

DOI: 10.2352/ISSN.2470-1173.2016.14.IPMVA-387

(a) Left stereo image (b) Right stereo image (c) CENSUS-dense (224 bits) (d) STABLE (64 bits)

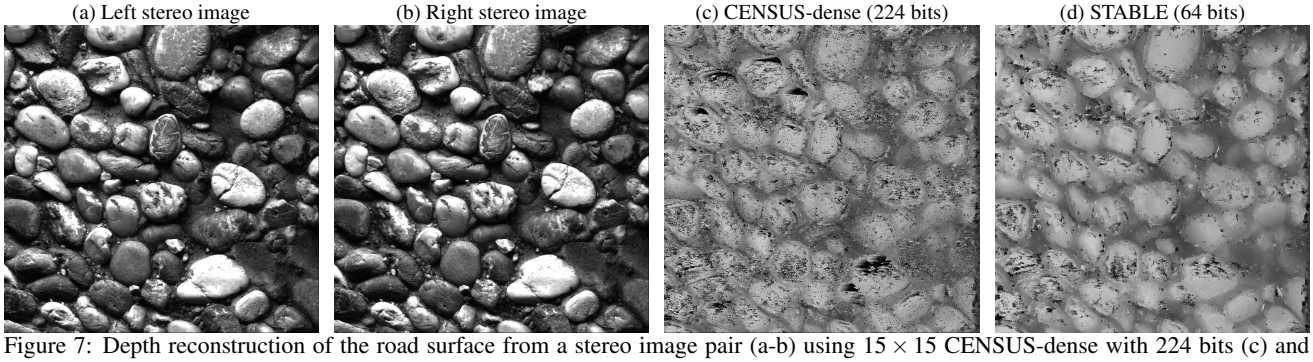

Figure 7: Depth reconstruction of the road surface from a stereo image pair (a-b) using 15×15 CENSUS-dense with 224 bits (c) and 15×15 STABLE with 64 bits (d).

(a) STABLE (16 bits) (b) STABLE (32 bits) (c) STABLE (64 bits) (d) STABLE (112 bits)

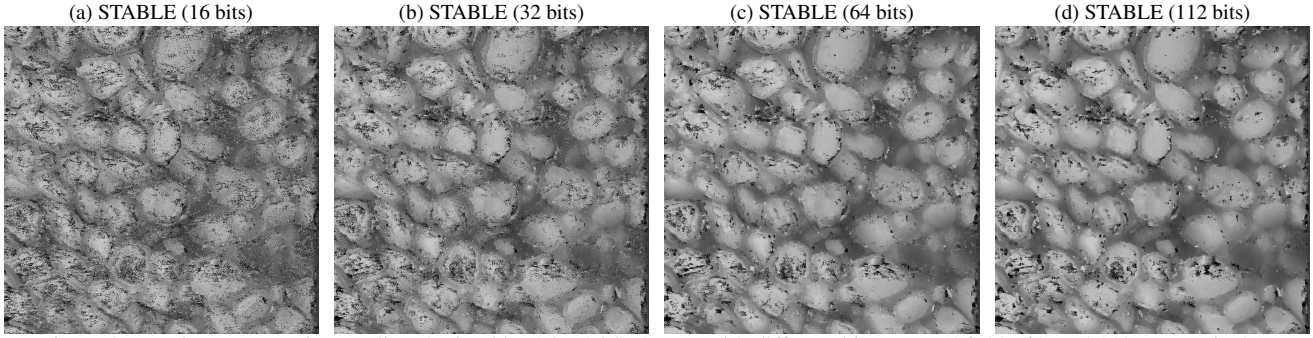

Figure 8: Depth reconstruction quality obtained by 15×15 STABLE with different bit counts (16, 32, 64, and 112, respectively).

(a) CENSUS-dense (224 bits); Input corrupted by 5×5 averaging (b) STABLE (64 bits); Input corrupted by 5×5 averaging (c) CENSUS-dense (224 bits); Input corrupted by -20 dB noise (d) STABLE (64 bits); Input corrupted by -20 dB noise

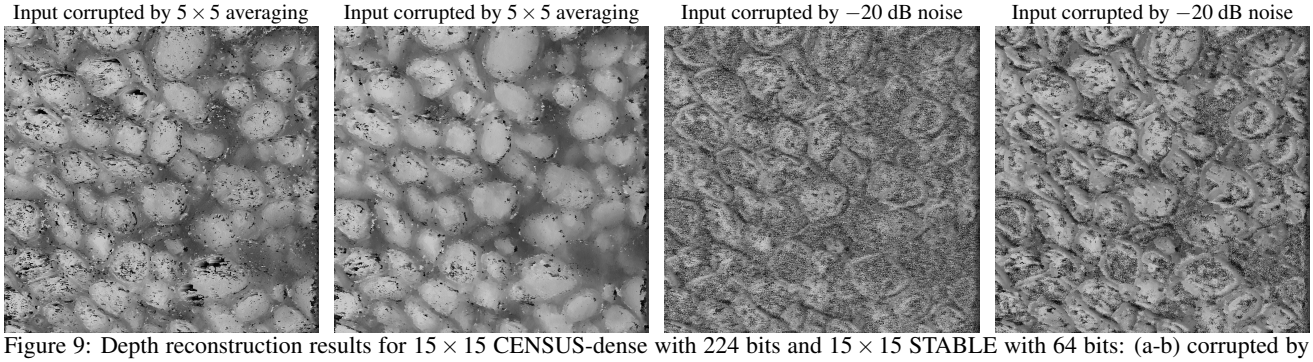

Figure 9: Depth reconstruction results for 15×15 CENSUS-dense with 224 bits and 15×15 STABLE with 64 bits: (a-b) corrupted by 5×5 averaging, (c-d) corrupted by Gaussian noise with -20 dB variance.