DEEP LEARNING FOR LOW TEXTURED IMAGE MATCHING

V. V. Kniaz[a,b]*, V. V. Fedorenko[a], N. A. Fomin[a]

[a] State Res. Institute of Aviation Systems (GosNIIAS), 125319, 7, Victorenko str., Moscow, Russia
(vl.kniaz, vfedorenko, nfomin73)@gosniias.ru
[b] Moscow Institute of Physics and Technology (MIPT), Russia

Commission II, WG II/8

KEY WORDS: image matching, deep convolutional neural networks, auto-encoders, cultural heritage

ABSTRACT:

Low-textured objects pose challenges for an automatic 3D model reconstruction. Such objects are common in archeological applications of photogrammetry. Most of the common feature point descriptors fail to match local patches in featureless regions of an object. Hence, automatic documentation of the archeological process using Structure from Motion (SfM) methods is challenging. Nevertheless, such documentation is possible with the aid of a human operator. Deep learning-based descriptors have outperformed most of common feature point descriptors recently. This paper is focused on the development of a new Wide Image Zone Adaptive Robust feature Descriptor (WIZARD) based on the deep learning. We use a convolutional auto-encoder to compress discriminative features of a local path into a descriptor code. We build a codebook to perform point matching on multiple images. The matching is performed using the nearest neighbor search and a modified voting algorithm. We present a new “Multi-view Amphora” (Amphora) dataset for evaluation of point matching algorithms. The dataset includes images of an Ancient Greek vase found at Taman Peninsula in Southern Russia. The dataset provides color images, a ground truth 3D model, and a ground truth optical flow. We evaluated the WIZARD descriptor on the “Amphora” dataset to show that it outperforms the SIFT and SURF descriptors on the complex patch pairs.

1. INTRODUCTION

Dense and robust image matching is a crucial step for an accurate 3D object reconstruction. Various kind of textures requires different feature descriptors for high-quality image matching. Often it is not easy to choose the best feature descriptor from the wide range of algorithms available nowadays. Each handcrafted descriptor has advantages and disadvantages. Usually, it achieves the best performance only for a specific kind of object’s texture. Recently a new generation of feature descriptors based on the deep learning was developed. Such descriptors can be trained for image matching for a given kind of object. Such specificity in the pair selection provides a dramatic increase in the matching precision and recall. The main disadvantage of the deep learning based descriptors is that the initial step of a dataset generation is necessary. The process of an accurate image path pair extraction can be time-consuming, and it is required for any new kind of object texture that was not present in the original training dataset. This paper is focused on the development of a new Wide Image Zone Adaptive Robust feature Descriptor (WIZARD) based on the deep learning. The paper presents two main contributions: (1) technique for on-the-fly automatic training dataset generation from the input imagery, (2) a new deep convolutional auto-encoder architecture (WIZARD) for the descriptor code generation.

The WIZARD descriptor is based on a convolutional auto-encoder for retrieval of the discriminative image features. The architecture is designed to work with the color images. It is based on the previous work for image matching in the infrared range (Knyaz et al., 2017). The auto-encoder takes a color image patch of 64×64 pixels as an input x. The image is processed by three convolutional layers to obtain a feature code F. The feature code is used as the WIZARD descriptor’s value for image matching. During the training stage, the auto-encoder is trained to compress the input x to code F and to recover back the input image from the code F to output image y. The loss function is based on the difference between the output image y and the input x. As the dimension of the code F is small (196 values) the encoder learns to compress the most discriminative features of the input image to code F. Hence, the code can be used efficiently for image patch matching (Kehl et al., 2016).

We perform on-the-fly generation of the codebook to adopt the auto-encoder to the target object texture domain. Firstly, image feature points are detected using FAST (Rosten and Drummond, 2006) or SUSAN (Smith and Brady, 1997) feature detectors. To generate the dataset small image patches are extracted for each detected feature point. After that, each image patch is applied as a texture to a random surface. The textured surface is rendered from various angles to new image patches that are used for codebook generation. All rendered images are collected in the training dataset to provide surface invariant image matching. Matching is performed using feature codes generated by an auto-encoder. A majority vote of feature codes for a given point defines its class. The 3D model generation pipeline is presented in figure 1.

2. RELATED WORK

Accurate image patch matching is required in many applications of photogrammetry and computer vision. For example, local patch descriptors must provide high robustness for an accurate 3D model generation using structure-from-motion (SfM) algorithms (Remondino et al., 2014, Knyaz et al., 2017). Nowadays, local patch descriptors became the main approach for robust point matching. In contrast with methods that perform dense point matching of the whole image (optical flow estimation (Farnebäck, 2003, Sun

*Corresponding author

https://doi.org/10.5194/isprs-archives-XLII-2-513-2018

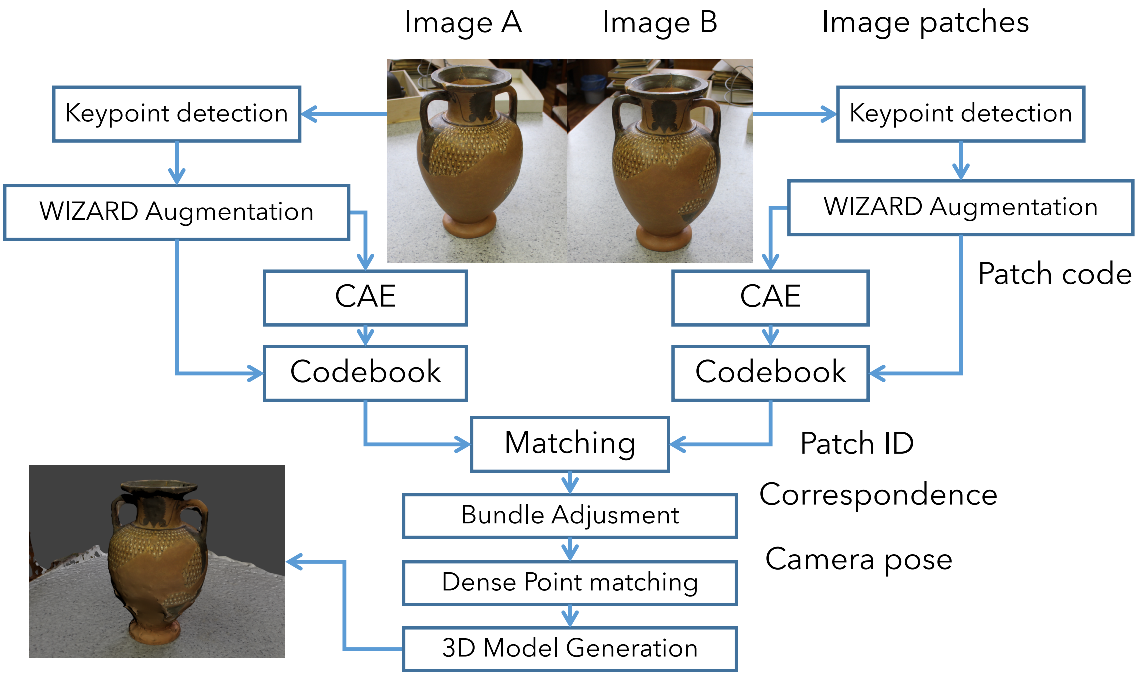


Figure 1. A 3D model generation pipeline

et al., 2010, Dosovitskiy et al., 2015), LSD-SLAM (Engel et al., 2014)), descriptors are designed for matching of small regions of an image a keypoint (edge, blob, etc.).

First approaches to point matching were based on the corner detection. Corner detectors proposed by Harris (Harris and Stephens, 1988) and Förstner (Förstner and Gülch, 1987) provided a degree of automation for photogrammetric software. The problem of matching of non-corner points has stimulated the development of blob descriptors such as Local Binary Patterns (LBP) (Wang and He, 1990) and the Census Transform (Zabih and Woodfill, 1994). Still, scale and rotation invariant descriptors were required to perform feature point matching “in the wild.” The SIFT descriptor (Lowe, 1999, Lowe, 2004) provided the desired degree of robustness to changes in an object’s pose and orientation. The high accuracy of the SIFT made it very popular in practical applications. It remains widely used for point matching in photogrammetry after nearly twenty years since the original paper.

The main disadvantage of the SIFT is high computational complexity. Multiple descriptors were proposed (SURF (Bay et al., 2008), ORB (Rublee et al., 2011), FREAK (Alahi et al., 2012)) that outperformed the SIFT in the processing time. However, the benefit in the time came at the cost of the matching quality. By the end of the 2010s, there was a vast list of handcrafted feature descriptors. Each descriptor had its benefits and disadvantages, that should be considered for an effective application.

The situation changed with the appearance of a new machine learning-based generation of feature descriptors in the 2010s. The new descriptors used the deep Convolutional Neural Networks (CNN) for an effective encoding of discriminative features. The modern GPUs reduced the computational time of the CNN-based descriptors, while training on the dedicated dataset provided a superior performance in the point matching on the target object.

Simo-Serra et al. (Simo-Serra et al., 2015) proposed a deep convolution feature point descriptors (DCFPD) that outperformed the SIFT on the Multi-view Stereo (Goesele et al., 2007) dataset. The DCFPD is based on a Siamese network that takes two images patches as an input. In the output, the network produces a similarity measure between the patches.

An alternative approach was proposed by Kehl et al. (Kehl et al., 2016) for matching of local RGB-D patches. The approach is based on a convolutional auto-encoder (CAE) (Goodfellow et al., 2016). In previous research (Knyaz et al., 2017) performed by authors, the CAE-based approach was successfully modified for matching local image patches in infrared images. The present paper is focused on the modification of the developed method for effective matching of the RGB local patches in low-textured images.

3. METHOD

Local patch matching with the WIZARD descriptor includes several steps. Firstly, a convolutional auto-encoder is trained offline. Secondly, the trained encoder is used to generate a codebook for every photo. Finally, the points are matched using the codebook. The following section presents details of each step.

3.1 Convolutional auto-encoders

An auto-encoder (AE) can be considered as a special case of a feed-forward neural network (Goodfellow et al., 2016). An AE consists of two parts: an encoder and a decoder. The encoder takes an input x and compresses it to a low dimensional code f. The decoder takes the code f as input and reconstructs the original signal in its output y. The objective function is the cross-entropy loss (logistic loss) for the input x and the reconstruction y.

A convolutional auto-encoder (CAE) is designed to process the images. The CAE uses convolutional layers in the encoder part. Image reconstruction is performed using deconvolutional layers.

Layer	Size out	Kernel	Stride
Input	$3 \times 64 \times 64$		
Convolution	$32 \times 31 \times 31$	2×2	4
Convolution	$32 \times 15 \times 15$	2×2	2
Convolution	$1 \times 14 \times 14$	2×2	2
Code	$1 \times 14 \times 14$		
Deconvolution	$32 \times 30 \times 30$	4×4	2
Deconvolution	$32 \times 62 \times 62$	4×4	2
Deconvolution	$3 \times 64 \times 64$	3×3	1

Table 1. CAE network architecture

The low-dimensional code f must contain the discriminative features of an input x to provide a high-quality reconstruction. Hence, the code can be used as a descriptor for a local patch matching.

A CAE architecture proposed in the previous research (Knyaz et al., 2017) was used as a starting point. Three contributions were made: (1) a convolutional layer and a deconvolutional layer were added to process a larger input patch, (2) two fully-connected layers were removed, (3) the number of the input's channels was increased to three. The resulting CAE architecture is presented in figure 2 and table 1.

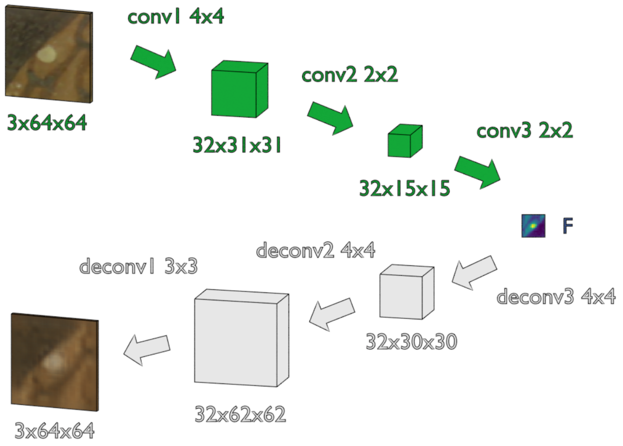


Figure 2. The convolutional auto-encoder architecture of the WIZARD descriptor

3.2 Dataset design

We present a new "Multi-view Amphora" (Amphora) dataset for evaluation of point matching algorithms. The dataset includes images of an Ancient Greek vase found at Taman Peninsula in Southern Russia. The dataset provides color images, a ground truth 3D model, and a ground truth optical flow. The dataset is publicly available[1].

The ground truth 3D model was generated using a fringe projection scanner (Knyaz, 2010). The external orientation of the camera was found for each color photo. The model was imported to the Blender 3D creation suite to create the ground truth optical flow for color images. The optical flow (figure 3) can be used to validate the matching results during the test stage.

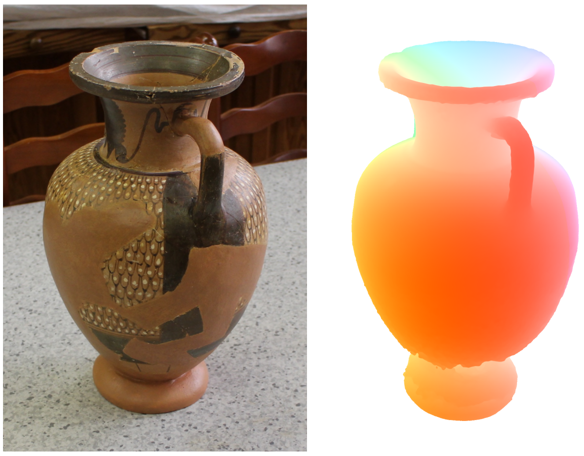


Figure 3. An image from the dataset and the corresponding ground truth optical flow

3.3 WIZARD codebook generation

A robust feature point descriptor must be invariant to changes in scale and rotation of the local patch. The codebook augmentation is used to address this aspect of the point matching. Firstly, a local path is extracted for each point detected on the input images (figure 4). After that, the local patch is projected on various surfaces: flat plane, sphere, corner, pit, etc.

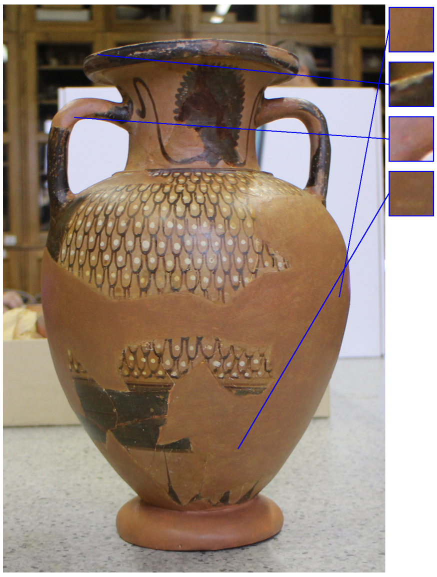
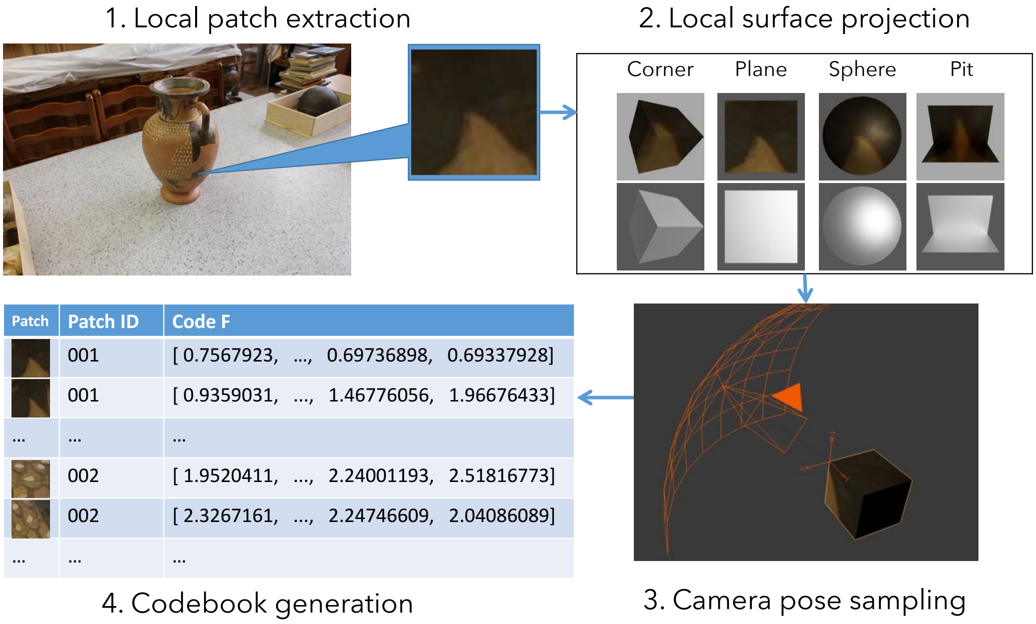


Figure 4. Feature point extraction

Multiple images are rendered using the textured model. The positions of a virtual camera are sampled randomly over a sector of a sphere (see figure 5). Multiple roll angles are selected randomly for each position. Positions and intensities of a virtual light source are also sampled randomly. Figure 5 summarized the complete augmentation process.

After the augmentation, all synthesized image patches are processed using the CAE to obtain the codebook. The codebook defines the correspondence between a point ID on the current image and the code F for point's local patch. For a single image the codebook includes $N \times g$ rows, where N is the number of

[1] http://www.zefirus.org/MVA18



Patch	Patch ID	Code F
	001	[0.7567923, ..., 0.69736898, 0.69337928]
	001	[0.9359031, ..., 1.46776056, 1.96676433]
...	...	...
	002	[1.9520411, ..., 2.24001193, 2.51816773]
	002	[2.3267161, ..., 2.24746609, 2.04086089]
...	...	...

Figure 5. Codebook augmentation

points detected by a feature detector, g is the number of patches synthesized during the augmentation.

3.4 Local patch matching

We perform local patch matching using a modified majority vote approach (Knyaz et al., 2017). Let $\boldsymbol{I}_q$ be a local image patch on the query image $\boldsymbol{Q}$. Let z_q be the local ID of the patch on the image $\boldsymbol{Q}$. Let C_b be the codebook for image $\boldsymbol{B}$ in which the corresponding patch $\boldsymbol{I}_b$ must be found. Firstly, we calculate the query patch code $F_q = CAE(\boldsymbol{I}_q)$. To find the corresponding patch, we query k IDs of nearest neighbors Z_b from the codebook C_b

$$Z_b = NearestNeighbors(C_b, z_q) = \{z_b^{j1}, z_b^{j1}, z_b^{j2}, \ \ldots\}, \tag{1}$$

The ID of the corresponding patch z_b is given by the patch ID of a majority vote of its neighbors. We define the probability that patch $\boldsymbol{I}_b^j$ corresponds to the patch $\boldsymbol{I}_q$ as follows

$$p = P(z_q = z_b^j) = \frac{|\{z_b \in Z_b : z_q = z_b^j\}|}{k}. \tag{2}$$

The probability p is used as a distance measure to filter the poor matches.

4. EVALUATION

We evaluate the WIZARD descriptor using the “Amphora” dataset. The following section presents details on training and evaluation of the descriptor.

4.1 CAE training details

CAE networks are trained in a semi-supervised setting. The training dataset does not provide any information about the correct patch pars. Hence, local patches for the dataset can be sampled using a sliding window. We created the dataset with 20000 local patches using images from the “Amphora” dataset. The training was performed using the PyTorch library (Paszke et al., 2017) and the NVIDIA 1080 GPU. The CAE networks have a moderate number of trained parameters compared to deep modern networks. Therefore the training process took only 30 minutes and 90 epochs. The CAE reconstruction results for the first and the last epochs are presented in figure 6.

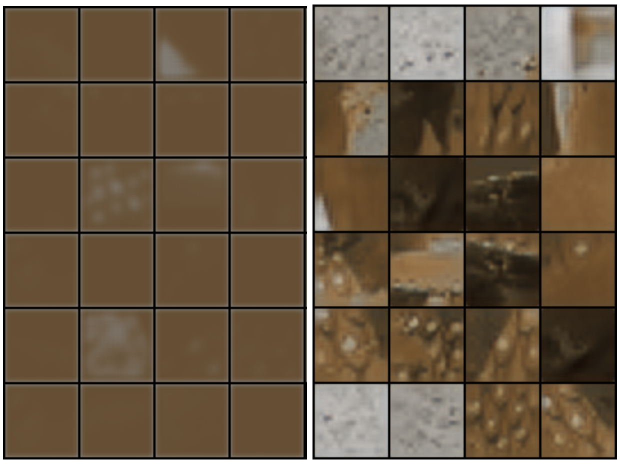


Figure 6. Examples of CAE reconstruction for the first (left) and the last (right) epochs

4.2 Comparison with other feature descriptors

We compared the WIZARD descriptor with the SIFT and the SURF feature descriptors. We used pairs of images from the “Amphora” dataset and the corresponding optical flow. The optical flow $\boldsymbol{U}(x, y)$ defines the displacement vector from the second image in the pair to the first image.

For each point on the second (query) image $\boldsymbol{Q}$, we searched the best match on the first image $\boldsymbol{B}$. We calculated the difference

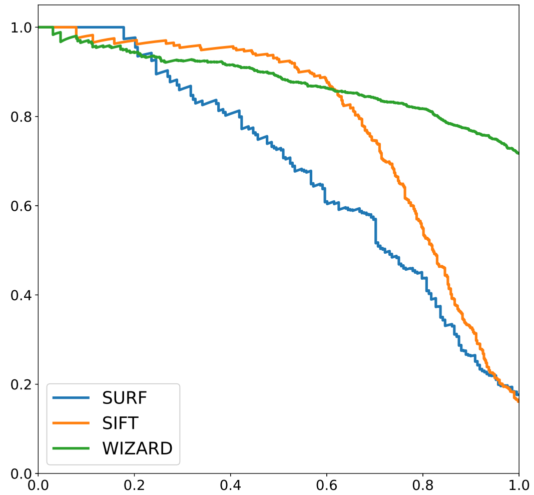


Figure 7. Precision-Recall curves for the SIFT, the SURF, and the WIZARD descriptors

between a location of the patch $\boldsymbol{x}_q$ on the query image, and the corresponding location $\boldsymbol{x}_b$ found by a descriptor

$$\hat{\boldsymbol{U}}(\boldsymbol{x}_q) = \boldsymbol{x}_q - \boldsymbol{x}_b. \tag{3}$$

The matching point is considered correct if the L^2 norm of the difference of the optical flow $\boldsymbol{U}$ and the calculated displacement $\hat{\boldsymbol{U}}$ is less than a threshold t. The decision function $d(\boldsymbol{U}, \hat{\boldsymbol{U}})$ returns one if the correspondence is correct, and zero otherwise

$$d(\boldsymbol{U}, \hat{\boldsymbol{U}}) = \begin{cases} 1, & ||\boldsymbol{U} - \hat{\boldsymbol{U}}||_2 < t \\ 0, & \text{otherwise} \end{cases}. \tag{4}$$

We use the probability p as the score value for the WIZARD descriptor. We use the difference between the best match distance and the second match distance as the score for the SIFT and the SURF descriptors

$$s(F_q, F_1, F_2) = \frac{||F_q - F_2||_2 - ||F_q - F_1||_2}{||F_q - F_2||_2}, \tag{5}$$

where F_q – is the descriptor's code for the query image, F_1 – is the code for the best match, F_2 – is the code for the second match. The Precision-Recall (PR) curves for the SIFT, the SURF, and the WIZARD descriptors are presented in figure 7.

The area under the PR curve for the WIZARD was 0.88, for the SIFT it was 0.79, and for the SURF it was 0.62. The analysis of the PR curve has shown that the WIZARD descriptor has outperformed the SIFT on similar pairs with high recall.

4.3 Applications

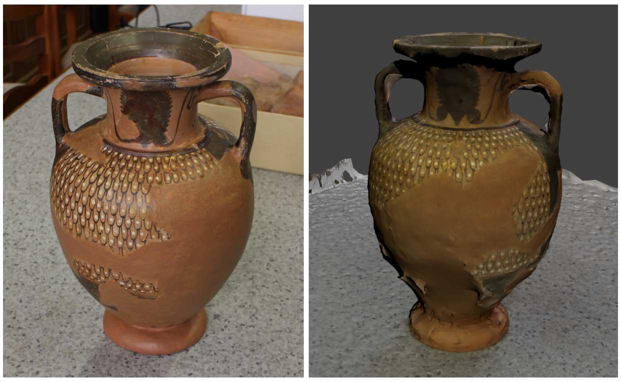


Figure 8. An example of 3D model reconstruction of an ancient vase using matches found by the WIZARD descriptor

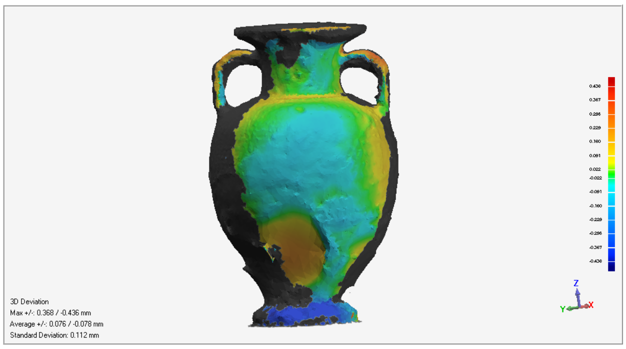


Figure 9. Accuracy evaluation for the reconstructed model

We evaluate the performance of the WIZARD descriptor in the full pipeline for a 3D model generation. The WIZARD descriptor was applied to find the corresponding image points on the images of the amphora. The final 3D object reconstruction was performed with the Agisoft PhotoScan using the point pairs found using the WIZARD descriptor. The result of the 3D reconstruction on the ancient vase is presented in figure 8. A comparison with the 3D models generated by original PhotoScan pipeline has shown that the WIZARD descriptor improves the surface accuracy in low-textured areas, e.g., bare clay areas of the vase (figure 9).

5. CONCLUSION

The new WIZARD deep learning-based feature descriptor has been developed. The descriptor uses a convolutional auto-encoder network to extract discriminative features from image patches effectively. The final matching is performed using a modified voting algorithm. A dataset was generated to evaluate the WIZARD descriptor. The dataset includes 39 images of an ancient amphora and the corresponding ground truth optical flow. The optical flow can be used for automatic evaluation of point matching algorithms. The evaluation of the WIZARD descriptor on the "Amphora" dataset has shown that it effectively outperforms the SIFT and the SURF descriptors in the matching accuracy of low-textured images patches.

ACKNOWLEDGEMENTS

The reported study was funded by Russian Foundation for Basic Research (RFBR) according to the research project N$^{\text{o}}$ 17-29-04410.

REFERENCES

Alahi, A., Ortiz, R. and Vandergheynst, P., 2012. Freak: Fast retina keypoint. In: *Computer vision and pattern recognition (CVPR), 2012 IEEE conference on*, Ieee, pp. 510–517.

Bay, H., Ess, A., Tuytelaars, T. and Van Gool, L., 2008. Speeded-up robust features (surf). *Computer vision and image understanding* 110(3), pp. 346–359.

Dosovitskiy, A., Fischery, P., Ilg, E., usser, P. H., Hazirbas, C., Golkov, V., van der Smagt, P., Cremers, D. and Brox, T., 2015. FlowNet: Learning Optical Flow with Convolutional Networks. *2015 IEEE International Conference on Computer Vision (ICCV)* pp. 2758–2766.

Engel, J., Schöps, T. and Cremers, D., 2014. LSD-SLAM: Large-Scale Direct Monocular SLAM. *ECCV* 8690(Chapter 54), pp. 834–849.

Farnebäck, G., 2003. Two-Frame Motion Estimation Based on Polynomial Expansion. *SCIA* 2749(Chapter 50), pp. 363–370.

Förstner, W. and Gülch, E., 1987. A fast operator for detection and precise location of distinct points, corners and centres of circular features. In: *Proc. ISPRS intercommission conference on fast processing of photogrammetric data*, pp. 281–305.

Goesele, M., Snavely, N., Curless, B., Hoppe, H. and Seitz, S. M., 2007. Multi-view stereo for community photo collections. In: *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, IEEE, pp. 1–8.

Goodfellow, I., Bengio, Y. and Courville, A., 2016. *Deep Learning*. MIT Press.

Harris, C. and Stephens, M., 1988. A Combined Corner and Edge Detector. In: *Alvey Vision Conference 1988*, Alvey Vision Club, pp. 23.1–23.6.

Kehl, W., Milletari, F., Tombari, F., Ilic, S. and Navab, N., 2016. Deep Learning of Local RGB-D Patches for 3D Object Detection and 6D Pose Estimation. *ECCV* 9907(7), pp. 205–220.

Knyaz, V. A., 2010. Multi-media projector single camera photogrammetric system for fast 3d reconstruction. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XXXVIII-5, pp. 343–348.

Knyaz, V. A., Vygolov, O., Kniaz, V. V., Vizilter, Y., Gorbatsevich, V., Luhmann, T. and Conen, N., 2017. Deep Learning of Convolutional Auto-Encoder for Image Matching and 3D Object Reconstruction in the Infrared Range. In: *The IEEE International Conference on Computer Vision (ICCV) Workshops*.

Lowe, D. G., 1999. Object Recognition from Local Scale-Invariant Features. In: *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*, IEEE Computer Society, Washington, DC, USA, pp. 1150–1150.

Lowe, D. G., 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* 60(2), pp. 91–110.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. and Lerer, A., 2017. Automatic differentiation in pytorch.

Remondino, F., Spera, M. G., Nocerino, E., Menna, F. and Nex, F., 2014. State of the art in high density image matching. *The Photogrammetric Record* 29(146), pp. 144–166.

Rosten, E. and Drummond, T., 2006. Machine Learning for High-Speed Corner Detection. In: *Computer Vision – ECCV 2006*, Springer, Berlin, Heidelberg, Berlin, Heidelberg, pp. 430–443.

Rublee, E., Rabaud, V., Konolige, K. and Bradski, G., 2011. Orb: An efficient alternative to sift or surf. In: *Computer Vision (ICCV), 2011 IEEE international conference on*, IEEE, pp. 2564–2571.

Simo-Serra, E., Trulls, E., Ferraz, L., Kokkinos, I., Fua, P. and Moreno-Noguer, F., 2015. Discriminative Learning of Deep Convolutional Feature Point Descriptors. *ICCV* pp. 118–126.

Smith, S. M. and Brady, J. M., 1997. SUSAN—A New Approach to Low Level Image Processing. *International Journal of Computer Vision* 23(1), pp. 45–78.

Sun, D., Sudderth, E. B. and Black, M. J., 2010. Layered image motion with explicit occlusions, temporal consistency, and depth ordering. *NIPS* pp. 2226–2234.

Wang, L. and He, D.-C., 1990. Texture classification using texture spectrum. *Pattern Recognition* 23(8), pp. 905–910.

Zabih, R. and Woodfill, J., 1994. Non-parametric local transforms for computing visual correspondence. In: *European conference on computer vision*, Springer, pp. 151–158.

https://doi.org/10.5194/isprs-archives-XLII-2-513-2018 | © Authors 2018. CC BY 4.0 License.