EE3-25 Deep Learning (2018-2019): Coursework - Interim Report

Martin Ferianc
mf2915@ic.ac.org, CID:00984924

Abstract

This work investigates denoising and describing images by using local descriptors in the N-HPatches dataset, a noisy adaptation of Homography patches (HPatches) dataset [1]. The baseline approach explores a shallow U-Net [2] in combination with L2-Net [3] to process a noisy image to a set of descriptors in sequence. The baseline approach achieved mean scores of 0.42 mAP on the patch retrieval, 0.61 mAP on the patch verification and 0.107 mAP on the image matching respectively.

1. Introduction

1.1. Problem

The task in this work is to perform image denoising and descriptor generation based on the noisy version of HPatches (Homography patches) dataset [1], named N-HPatches with the use of deep learning. The generalisation performance is being evaluated by using three separate metrics: patch verification, image matching and patch retrieval.

Patch verification measures the ability of a descriptor to classify whether two patches are extracted from the same measurement. Image matching tests to what extent a descriptor can correctly identify correspondences in two images. Patch retrieval tests how well a descriptor can match a query patch to a pool of patches extracted from many images, including distractors [4].

1.2. Dataset

The N-HPatches dataset and its accompanying benchmark are primarily meant for evaluation of local descriptors in images. The size of black and white images is 32×32 to which three levels of increasing noise values, together with affine transformations, were applied. The prior over the types of noise is unknown. There are 31,179 and 19,050 images for denoising for training and validation respectively and 2,000 and 200 triplets split across training and validation for the descriptors of the initially noisy images.

2. Baseline Approach

The baseline approach consists of two sequentially connected models: for image denoising and for generation of the descriptors. The denoising network uses a bottom part of UNet, which has been originally used for image segmentation [2], giving 59,777 trainable parameters. The descriptor network is a L2-Net [3], with 1,336,032 trainable parameters, which was developed to extract local feature from images, where the output descriptor can be matched in Euclidean space by L2 distance.

Both networks are being trained by stochastic gradient descent for one epoch as seen in Figures 2 and 3, which does not adapt its parameters after the weight update. Additionally, the denoising network uses Nesterov momentum to accelerate the gradient descent. The denoising network uses mean absolute error loss, while the descriptor network is using triplet loss [5], where weights are shared across three identical networks.

2.1. Evaluation

The proposed denoising network is rather shallow and it does not include any regularisation. Looking at the losses in Figures 2 and 3, it is deemed that after more epochs they cannot significantly improve and they will overfit over time. Although, the descriptor network uses batch normalisation [6] for regularisation.

After the training, the Figure 1 shows that noise has been removed, however the proportions of different frequencies of noise being removed are inadequate, which suggests that the improved approach could aim to balance noise removal of different types. The exact results can be seen in Tables 1, 2 & 3. Overall, the baseline approach achieved mean scores of **0.42** mAP on the patch retrieval, **0.61** mAP on the patch verification and **0.107** mAP on the image matching. It is difficult to draw individual conclusions on each network as the networks are connected and dependent on each other.

3. Improved approach

The improved approach should still consist of two separate networks with larger capacity connected in series and be trained by using an advanced optimiser such as Adam [7] for several epochs with stopping conditions. Both, the improved denoising and descriptor networks could be regularised residual networks [8] which would gradually decompose the input into hierarchical features, while making training less demanding and more transparent.

The hyper-parameters: the capacity of the network in number of filters, their depth together with the amount of attention per noise type could be varied. The best candidates would be found through training on a smaller dataset and observing the training and validation losses as well as the respective performance on the outlined tasks.

References

[1] V. Balntas, K. Lenc, A. Vedaldi, and K. Mikolajczyk, "Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors," in *CVPR*, 2017.

[2] O. Ronneberger, P.Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, vol. 9351 of *LNCS*, pp. 234–241, Springer, 2015. (available on arXiv:1505.04597 [cs.CV]).

[3] Y. Tian, B. Fan, and F. Wu, "L2-Net: Deep learning of discriminative patch descriptor in euclidean space," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6128–6136, July 2017.

[4] A. Barroso and A. Lopez, "Keras triplet descriptor," 2019.

[5] D. P. Vassileios Balntas, Edgar Riba and K. Mikolajczyk, "Learning local feature descriptors with triplets and shallow convolutional neural networks," in *Proceedings of the British Machine Vision Conference (BMVC)* (E. R. H. Richard C. Wilson and W. A. P. Smith, eds.), pp. 119.1–119.11, BMVA Press, September 2016.

[6] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pp. 448–456, JMLR.org, 2015.

[7] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015.

A. Appendix

The instructions to run the code are in the `README.md` *in the zip package.*

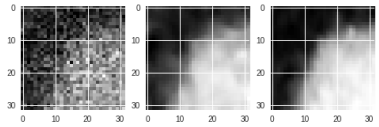

Figure 1: Denoising with the baseline method. On the left is a noisy patch, in the middle is the denoised patch and on the right is the reference image.

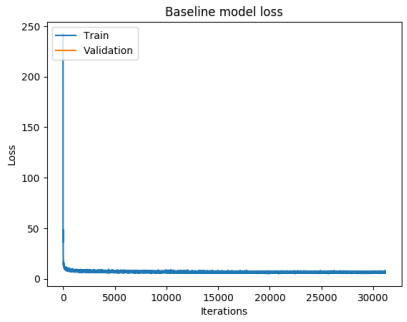

Figure 2: Loss for the denoising network.

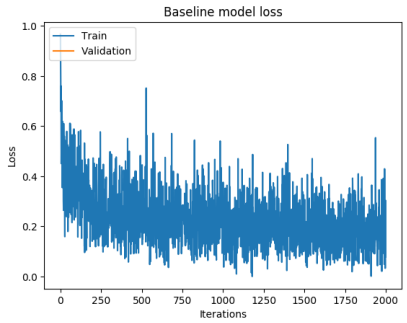

Figure 3: Loss for the descriptor network.

Baseline Approach	**Balanced Variant [AUC]**		**Imbalanced Variant [mAP]**		**Improved Approach**	**Balanced Variant [AUC]**		**Imbalanced Variant mAP**	
Noise	**Inter**	**Intra**	**Inter**	**Intra**	**Noise**	**Inter**	**Intra**	**Inter**	**Intra**
Easy	0.929	0.897	0.828	0.742	**Easy**				
Hard	0.910	0.870	0.761	0.646	**Hard**				
Tough	0.880	0.834	0.678	0.551	**Tough**				

Table 1: Verification Results.

Baseline Approach								**Improved Approach**							
	100	**500**	**1000**	**5000**	**10000**	**15000**	**20000**		**100**	**500**	**1000**	**5000**	**10000**	**15000**	**20000**
Easy [mAP]	0.745	0.609	0.552	0.434	0.386	0.361	0.345	**Easy [mAP]**							
Hard [mAP]	0.675	0.488	0.412	0.264	0.215	0.190	0.175	**Hard [mAP]**							
Tough [mAP]	0.574	0.355	0.278	0.148	0.111	0.094	0.084	**Tough [mAP]**							
Mean [mAP]	0.665	0.484	0.414	0.282	0.237	0.215	0.201	**Mean [mAP]**							

Table 2: Retrieval Results.

Baseline Approach				**Improved Approach**			
Easy [mAP]	**Hard [mAP]**	**Tough [mAP]**	**Mean [mAP]**	**Easy [mAP]**	**Hard [mAP]**	**Tough [mAP]**	**Mean [mAP]**
0.213	0.079	0.028	0.107				

Table 3: Matching Results.