ModuleNet: A Convolutional Neural Network for Stereo Vision

O. I. Renteria-Vidales[1,2], J. C. Cuevas-Tello[1], A. Reyes-Figueroa[2], and M. Rivera[2(✉)]

[1] UASLP Universidad Autónoma de San Luis Potosí, Álvaro Obregón 64 Col. Centro, 78000 San Luis Potosí, México
[2] CIMAT Centro de Investigación en Matemáticas A.C., Jalisco S/N Col. Valenciana, 36023 Guanajuato, México
mrivera@cimat.mx

Abstract. Convolutional Neural Networks (CNN) has gained much attention for the solution of numerous vision problems including disparities calculation in stereo vision systems. In this paper, we present a CNN based solution for disparities estimation that builds upon a basic module (BM) with limited range of disparities that can be extended using various BM in parallel. Our BM can be understood as a segmentation by disparity and produces an output channel with the memberships for each disparity candidate, additionally the BM computes a channel with the out–of–range disparity regions. This extra channel allows us to parallelize several BM and dealing with their respective responsibilities. We train our model with the MPI Sintel dataset. The results show that ModuleNet, our modular CNN model, outperforms the baseline algorithm Efficient Large-scale Stereo Matching (ELAS) and FlowNetC achieving about a 80% of improvement.

Keywords: Stereo vision · Convolutional Neural Networks · U-Net · Census transform · Deep learning

1 Introduction

The purpose of an stereo system is to estimate the scene depth by computing horizontal disparities between corresponding pixels from an image pair (left and right) and has been intensively investigated for several decades. There is a wide variety of algorithms to calculate these disparities that are complicated to include them all in one methodology or paradigm. Scharstein and Szeliski [13] propose a block taxonomy to describe this type of algorithms, following steps such as matching cost calculation, matching cost aggregation, disparity calculation and disparity refinement. One example is ELAS, an algorithm which builds a disparities map by triangulating a set of support points [8].

We present a CNN based solution for disparities estimation that builds upon a basic module (BM) with limited range of disparities that can be extended using various BM in parallel. Our BM can be understood as a segmentation

K. M. Figueroa Mora et al. (Eds.): MCPR 2020, LNCS 12088, pp. 219–228, 2020.
https://doi.org/10.1007/978-3-030-49076-8_21

by disparity and produces an output channel with the memberships for each disparity candidate, additionally the BM computes a channel with the out–of–range disparity regions. This extra channel allows us to parallelize several BM and dealing with their respective responsibilities. We list our main contributions as follows: i) We propose ModuleNet, which is a novel modular model to measure disparities on any range, which is inspired on FlowNet and U-Net. ii) We use a low computational time algorithm to measure cost maps. iii) The architecture of our model is simple, because it does not require another specialized networks for refinement as variants of FlowNet do for this problem. iv) Our model improves the baseline model ELAS and FlowNetC (the correlation version of FlowNet) with about 80% of unbiased error.

The paper is organized as follows: Sect. 2 presents the related work. At Sect. 2 are the algorithms FlowNet, Census transform and ELAS. The proposed model is in Sect. 3. Section 4 describes the dataset used in this research. At the end are our results, conclusions and future work.

2 Related Methods

In recent years, Convolutional Neural Networks (CNN) have made advances in various computer vision tasks, including estimation of disparities in stereo vision. Fischer et al. propose a CNN architecture based on encoder-decoder called FlowNet [6]. This network uses an *ad hoc* layer for calculating the normalized cross-correlation between a patch in the left image and a set of sliding windows (defined by a proposed disparity set) of the right window and uses Full Convolutional Network (kind encoder-decoder architecture) for estimate the regularized disparity [11]. Park and Lee [9] use a siamese CNN to estimate depth for SLAM algorithms. Their proposal is to train a twin network that transforms patches of images and whose objective is to maximize the normalized cross correlation between corresponding transformed patches and minimize it between non-corresponding transformed patches. To make the inference of the disparity in a stereo pair, a left patch and a set of displaced right patches are used, then the normalized cross correlation between the twin networks transformed patches and the disparity is selected using a Winner–Takes–All (WTA) scheme. Other authors use a multi-scale CNN, where the strategy is to estimate the disparity of the central pixel of a patch by processing a pyramid of stereo pairs [4]; and the reported processing time for images in the KITTI database is more than one minute [7]. A state of the art method with really good results is reported by Chen and Jung [3], they use a CNN that is fed with patches of the left image and a set of slipped patches of the right image (3DNet). Then, for a set of proposed disparities, the network estimates the probability that each of the disparities corresponds to the central pixel of the left image patch that requires of evaluate as many patches as pixels, so it is computationally expensive.

In this section, we present FlowNet, an architecture designed for optical flow, and it can be used for stereoscopy. Also, this section introduces the Census Transform.

2.1 FlowNet

FlowNet is composed by two main blocks. The network computes the local vector that measure the dissimilarity between each pixel (x, y) in the left image I_l and its corresponding candidate pixel $(x + \delta, y)$, for a given disparity δ, in the right image I_r; where $\delta \in d$ with $d = [d_1, d_2, \ldots, d_h]$ and d_i is an integer value. This block is deterministic (not trainable) and produces a dissimilarity map (tensor) D of size equal to $(h, nrows, ncolumns)$. FlowNet is based on the U-Net [11]. The network computes the regularized disparities d^*; with dimension equal to $(1, nrows, ncolumns)$. The main disadvantage of this method is the computational cost of the normalized cross-correlation layer and it also produces blurred disparity maps [6], see in Fig. 1 the FlowNetC architecture.

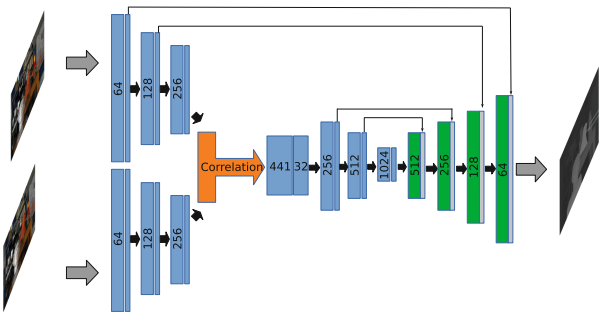

Fig. 1. FlowNet architecture.

2.2 Census Transform

Differently to FlowNet, that uses normalized cross-correlation to measure the cost maps, an alternative is Census Transform [15]. Other algorithms for this task are Sum of Absolute Differences (SAD) [14], Sum of Square Differences (SSD) [14], Normalized Cross-Correlation [5]. Because a low complexity cost function is desirable, we chose the Census Transform [15]. Figure 2 exemplify the Census algorithm, where it transforms the values of the neighbors. The values of the neighbors of a pixel are encoded within a binary chain (it is assigned "1" when they are greater than or equal to the central pixel, or "0" otherwise). This chain is called census signature, the signature retains spatial information of each neighbor given the position within the string where each bit is stored.

For a 3×3 window, the census signature contains eight values and can be saved in one byte, this transformation can be computed with:

$$C_l(x, y) = Bitstring_{(i,j)\in w}(I_l(i, j) \geq I_l(x, y)) \tag{1}$$

for the case of the left image I_l; and in a similar is computer the census transform C_r for the right image I_r. To obtain the level of correspondence, the Hamming

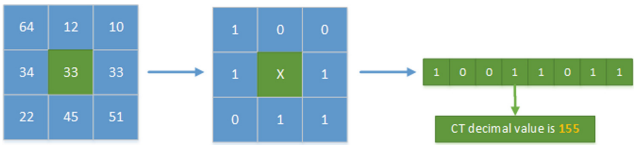

Fig. 2. Census transform

distance (H) is used to count how many bits are different between two census signatures:

$$D_m(x, y; d) = H(C_l(x, y), C_r(x + d_m, y)) \tag{2}$$

We can denote this stage by the representational function F_c that transforms the information in the images I_l and I_r into the distance tensor $D = [D_1, D_2, \ldots, D_h]$:

$$D = F_c(I_l, I_r; d) \tag{3}$$

where the parameters are the set of candidate disparities, d.

3 ModuleNet: Modular CNN Model for Stereoscopy

Our proposed model (ModuleNet) builds upon U-Net blocks and is inspired on the FlowNet. First, we describe the general block U-Net (see Fig. 3) that can find disparities in a range d. Second, we introduce the cascade U-Net for refinement, see Fig. 4. Finally, the modular CNN model (ModuleNet) for disparities out of the range d is presented, see Fig. 5.

3.1 General Block: U-Net U-Net Module

Our neural network for stereo disparity estimation is composed with blocks based on the UNet. Indeed, the most basic construction block can be seen as a simplified version of the FlowNet where the Disparity Map D is computed with the Hamming distances between the Census transformed patches (the fixed and the δ-displaced one). Another difference between our basic block and the FlowNet model is that, instead of computing directly a real valued map of disparities, we estimate the probability that a particular candidate disparity δ is the actual one at each pixel. We also compute an additional layer that estimates outliers: the probability that the actual disparity in each pixel is not included in the set of disparities d. As input to the U-Net, we have h channels of distances corresponding to the h candidate disparities and, as output, we have $h+1$ probability maps; see Fig. 3. We can represent this U-Net block by the representational function F_1 that transforms the information in the distance tensor $D = [D_1, D_2, \ldots, D_h]$ into the probabilities tensor $P = [P_1, P_2, \ldots, P_h, P_{h+1}]$:

$$P = F_1(D, \theta_1) \tag{4}$$

where θ_1 are the network weight set.

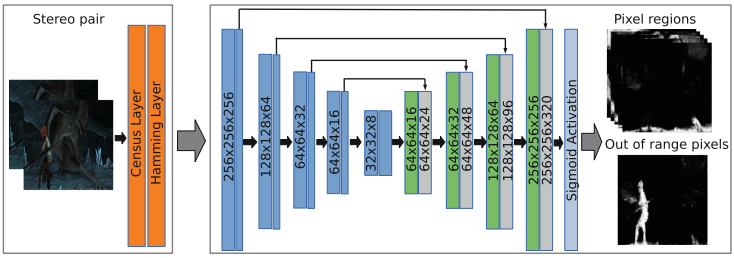

Fig. 3. General block (U-Net)

The representational U-Net F_1 (4) can be seen as a regularizer of the noisy Census-distance maps. We observed that the output of the basic (trained) block can be refined by a second U-Net. This second U-Net (in cascade) is trained using as input the census cost maps, the initial estimation of the disparity probabilities maps and the outliers' probability map and produces as output refined versions of the inputs. We represent this U-Net block by the representational function F_2 that refine probabilities tensor P using also as input the distance tensor D:

$$\hat{Y} = F_2(P, D, \theta_2) \tag{5}$$

where θ_2 are the weight set. We denote our basic module for disparity estimation by

$$D = F_c(I_l, I_r; d) \tag{6}$$

$$\hat{Y} = F(D) \stackrel{def}{=} F_2(F_1(D), D). \tag{7}$$

where we omitted the parameters θ_1 and θ_2 in order to simplify the notation. Once we have trained a basic module (7), it can be used for estimating disparities into the range defined by the disparity set d. The regions with disparities outside such a range are detected in the outliers' layer. Figure 4 depicts our block model based on two cascaded U-Nets (general blocks, see Fig. 3).

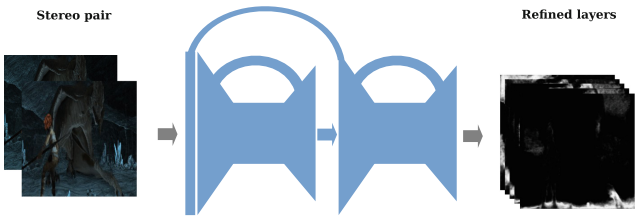

Fig. 4. Our Basic Block composed with two U-Net in cascade.

3.2 ModuleNet: Modular CNN Model

Assume, we have a trained basic module for the disparities into the interval $[d_1, d_h]$ and the actual range of disparities, in the stereo pair, lays into the interval $[d_1, 2d_h]$. We can reuse our basic model for processing of such a stereo pair if we split the calculations for the disparities sets $d^{(0)} = [d_1, d_2, \ldots, d_h]$ and $d^{(1)} = [d_{h+1}, d_{h+2}, \ldots, d_{2h}]$. Then, we can compute two census distance tensors $D^{(0)} = F_c(I_r, I_l; d^{(0)})$ and $D^{(1)} = F_c(I_r, S\{I_l, h\}; d^{(1)} - h)$; where we define the shift operator

$$S\{I, d_h\} \stackrel{def}{=} I(x + d_h, y). \tag{8}$$

Thus, we can estimate the probability that the disparity is in the set $d^{(0)}$ with $\hat{Y}^{(0)} = F(D^{(0)})$ and in the set $d^{(1)}$ with $\hat{Y}^{(1)} = F(D^{(1)})$; where F is our basic module 7.

This idea can be extended for processing stereo pair with a wide range of disparities. First we define the k-th set of disparities as

$$D^{(k)} = F_c(I_r, S\{I_l, kh\}; d^{(k)} - kh) \tag{9}$$

for $k = 1, 2, \ldots, K$. Second, we estimate, in parallel, the K tensor of probability:

$$\hat{Y}^{(k)} = F(D^{(k)}) \tag{10}$$

Note that the network F is reused for processing the K modules. The CNN transforms the representation $D^{(k)}$ into $\hat{Y}^{(k)}$: the probability that disparities $\delta^{(k)}$ of the module k at the pixel (x, y) are the correct displacement. To estimate the tensor $\hat{Y}$ that integrates the individual probability tensors $\hat{Y}^{(k)}$'s, we use the additional layer with the probability that the correct displacement of each pixel is not the k-th interval:

$$\hat{Y}_{(kh+i)} = \hat{Y}_i^{(k)} \odot \left(1 - \hat{Y}_{h+1}^{(k)}\right) \tag{11}$$

for $i = 1, 2, \ldots, h$, $k = 0, 1, \ldots, K-1$ and $\odot$ denotes the element-wise product. Finally, the disparity estimation, d^* is computed by applying a WTA procedure in the disparities map $\hat{Y}$:

$$d^*(x, y) = \arg\max_l \hat{Y}_l(x, y) \tag{12}$$

for $l = 1, 2, \ldots, Kh$. Figure 5 depicts ModuleNet – our modular model.

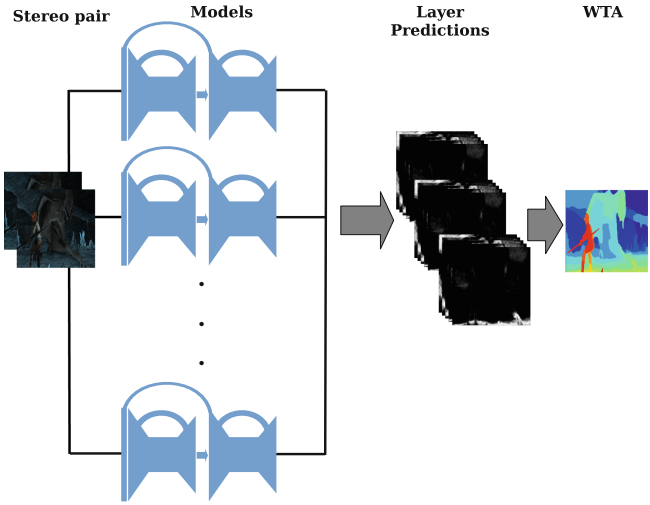

Fig. 5. ModuleNet: Modular CNN Model

4 Dataset and Training Parameters

We used the MPI Sintel dataset for train our model. The MPI Sintel-stereo dataset is a benchmark for stereo, produced from the open animated short film Sintel produced by Ton Roosendaal and the Blender Foundation [1]. This dataset contains disparity maps for the left and right image, occlusion masks for both images. The dataset consist of 2128 stereo pairs divided in clean and final pass images. The left frame is always the reference frame. For our experiments, we use the clean subset pairs that consist of 1064 pairs; 958 for training and 106 for testing. See example in Fig. 6, the disparity map is the ground truth. Our training set consisted on patches (256×256 pixels) randomly sampled from of 958 stereo pairs (1024×460 pixels) and 106 stereo pairs were leave-out for testing.

We change the number of filters distributions across the layers according to Reyes-Figueroa et al. [10]. It has been shown that in order to have more accurate features and to recover fine details, more filters are required in the upper levels of U-Net and less filters in the more encoded levels. Our model's architecture is summarized in Fig. 3. We trained our model during 2000 epochs with mini-batches of size eight.

We used data augmentation by randomly resizing the frames (random scaling factor into the range $[.6, 1]$), adding Gaussian noise (mean zero with standard deviation equal 1% the images' dynamic range). The ADAM optimization algorithm was used with fixed $lr = 1 \times 10^{-4}$ and $\beta = [0.9, 0.999]$. For processing the data set, we used a basic block with sixteen disparities ($h = 16$) and $K = 24$ parallels blocks.

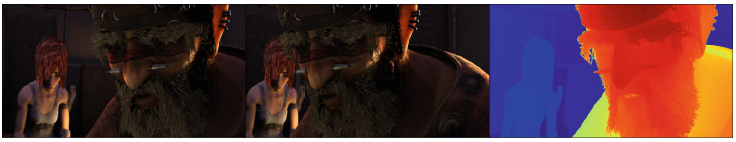

Fig. 6. Example of MPI Sintel data: left and right images and disparity map.

5 Results

In Fig. 7 are shown the results from seven scenes by using the MPI Sintel dataset. We show a single image per scene for illustrating the algorithm's performance. We compare the results from our model versus ELAS and FlowNetC. Visually, one can see that the proposed model is closer to the ground truth than ELAS and FlowNetC.

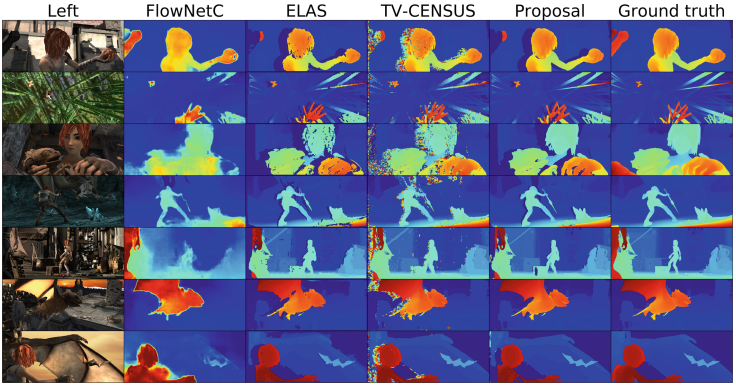

Fig. 7. Results from MPI Sintel dataset on selected scenes

In Table 1 is the comparison of results from applying a Total–Variation potential for edge–preserving filtering to the Distance Tensor D (here named TV–Census) [2], ELAS, FlowNetC and our proposal (ModuleNet); in bold font the best results. We use the metric Mean Absolute Error (MAE) in non-occluded areas to measure the results quantitatively. Our proposed model outperforms the compared methods. The advantage of the MPI Sintel dataset is that the ground truth is provided, so the accuracy (MAE) is unbiased. Show particular results from seven representative stereo pairs and the average over the total of frames. Additionally we tested our method with the Middlebury Stereo Datasets 2014 [12] which consist of 33 image pairs, divided in 10 evaluation test sets with hidden ground truth, 10 evaluation training sets with ground truth and 13 additional sets, the first 20 sets are used in the new Middlebury Stereo Evaluation. Figure 8 shows a visual comparison of the computed results.

Table 1. MAE results from MPI Sintel dataset on selected scenes

Scene	FlowNetC	ELAS	TV–Census	Proposed
alley_1	2.98	2.98	0.92	**0.44**
bamboo_1	2.91	2.39	0.63	**0.51**
bandage_2	14.09	12.77	2.60	**2.14**
cave_2	3.95	3.10	1.85	**0.65**
market_2	1.94	2.07	0.54	**0.43**
temple_2	2.26	2.44	0.60	**0.38**
temple_3	6.09	2.85	0.74	**0.43**
All test images	24.3	14.1	1.7	**1.5**

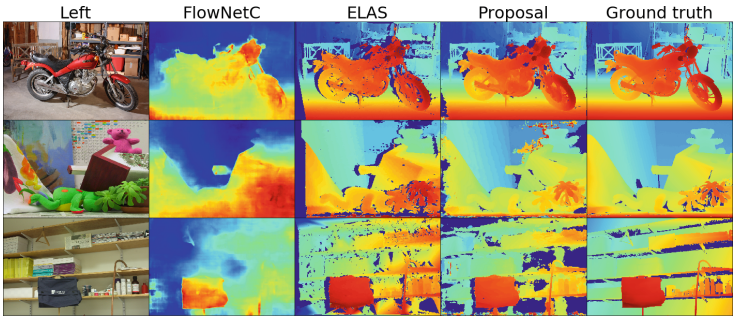

Fig. 8. Results from Middlebury dataset on selected stereo pairs

6 Conclusions and Future Work

We proposed a new model called ModuleNet for disparities estimation that can be applied in stereoscopy vision. Our model is build upon FlowNet, U-Net and Census transform. The modularity of our method allows generating disparity maps of any size simply by adding more blocks. The extra layer, for detecting pixels with disparities out of range, helps us to classify pixels that usually adds noise since these pixels are outside the range of work or are pixels of occluded regions. Our results show that qualitatively and quantitatively our model outperforms Census–Hamming approach (robustly filtered), ELAS and FlowNetC; which are baseline methods for disparities estimation. The unbiased error was improved by about 80%.

Our future work will focus on extend the training set with real stereo pairs, conduct more exhaustive evaluations and implement our model on an embedded system (e.g. NVIDIA® Jetson Nano™ CPU-GPU and Intel®Movidius™ USB stick). We plan to compare the performance of our model with other state-of-the-art methods, regardless the complexity and computational time with GPU hardware. As most of the methods, the texture-less regions are difficult to identify. So an algorithm to detect such textures is desired.

Acknowledges. Part of this work was conducted while O. Renteria was at IPICYT AC, SLP-Mexico. This work was supported in part by CONACYT, Mexico (Grant A1-S-43858).

References

1. Butler, D.J., Wulff, J., Stanley, G.B., Black, M.J.: A naturalistic open source movie for optical flow evaluation. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7577, pp. 611–625. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33783-3_44
2. Charbonnier, P., Blanc-Féraud, L., Aubert, G., Barlaud, M.: Deterministic edge-preserving regularization in computed imaging. IEEE Trans. Image Process. **6**(2), 298–311 (1997)
3. Chen, B., Jung, C.: Patch-based stereo matching using 3D convolutional neural networks. In: 25th ICIP, pp. 3633–3637 (2018)
4. Chen, J., Yuan, C.: Convolutional neural network using multi-scale information for stereo matching cost computation. In: ICIP, pp. 3424–3428 (2016)
5. Duda, R.O., Hart, P.E.: Pattern Classification and Scene Analysis, XVII, p. 482. Wiley, New York (1973)
6. Fischer, P., et al.: FlowNet: learning optical flow with convolutional networks. In: CoRR, pp. 2758–2766 (2015)
7. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. In: CVPR, pp. 3354–3361 (2012)
8. Geiger, A., Roser, M., Urtasun, R.: Efficient large-scale stereo matching. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010. LNCS, vol. 6492, pp. 25–38. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19315-6_3
9. Park, J., Lee, J.: A cost effective estimation of depth from stereo image pairs using shallow siamese convolutional networks. In: IRIS, pp. 213–217, October 2017
10. Reyes-Figueroa, A., Rivera, M.: Deep neural network for fringe pattern filtering and normalisation (2019). arXiv:1906.06224)
11. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
12. Scharstein, D., et al.: High-resolution stereo datasets with subpixel-accurate ground truth. In: Jiang, X., Hornegger, J., Koch, R. (eds.) GCPR 2014. LNCS, vol. 8753, pp. 31–42. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11752-2_3
13. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J. Comp. Vision **47**(1), 7–42 (2002). https://doi.org/10.1023/A:1014573219977
14. Tamura, H., Mori, S., Yamawaki, T.: Textural features corresponding to visual perception. IEEE Trans. Sys. Man Cybern. **8**, 460–473 (1978)
15. Zabih, R., Woodfill, J.: Non-parametric local transforms for computing visual correspondence. In: Eklundh, J.-O. (ed.) ECCV 1994. LNCS, vol. 801, pp. 151–158. Springer, Heidelberg (1994). https://doi.org/10.1007/BFb0028345