Innovation in Medicine and Healthcare 2014
M. Graña et al. (Eds.)
IOS Press, 2014

doi:10.3233/978-1-61499-474-9-74

Texture Descriptors Based on Dijkstra's Algorithm for Medical Image Analysis

Stefano GHIDONI[a], Loris NANNI[a,1], Sheryl BRAHNAM [b], and Emanuele MENEGATTI [a]
[a] *DEI, University of Padua, Padua, Italy*
[b] CIS, Missouri State University, Springfield, MO USA

Abstract. In this paper we investigate a new approach for extracting features from a texture using Dijkstra's algorithm. The method maps images into graphs and gray level differences into transition costs. Texture is measured over the whole image comparing the costs found by Dijkstra's algorithm with the geometric distance of the pixels. In addition, we compare and combine our new strategy with a previous method for describing textures based on Dijkstra's algorithm. For each set of features, a support vector machine (SVM) is trained. The set of classifiers is then combined by weighted sum rule. Combining the proposed set of features with the well-known local binary patterns and local ternary patterns boosts performance. To assess the performance of our approach, we test it using six medical datasets representing different image classification problems. Tests demonstrate that our approach outperforms the performance of standard methods presented in the literature. All source code for the approaches tested in this paper will be available at: http://www.dei.unipd.it/node/2357.

Keywords. texture descriptors, local binary patterns, ensemble, support vector machines

Introduction

Medical imaging is a field that has advanced considerably the last couple of decades. Even though medical images are commonly analyzed by human experts, the huge amount of data currently being produced has motivated researchers to explore the possibility of automating image analysis. This presents a challenge as the medical images currently available investigate phenomena that exhibit a great deal of variation, creating the need to develop a large set of different automatic analysis methods. Moreover, even though automatic analysis has not reached the quality levels achieved by manual analysis, it still offers many advantages, especially noteworthy being the capacity of analyzing and relating huge amounts of data. Most experts agree that this capability has the potentially of revolutionizing scientific knowledge in medicine.

The analysis of medical images often relies on texture, as the textural content of images includes a large amount of information that is more important in medicine than in other domains like robotics and video surveillance. For this reason, texture analysis methods are often employed in medical image analysis. Common texture analysis

---

[1] Corresponding Author: Loris Nanni, DEI, University of Padua, viale Gradenigo 6, Padua, Italy. loris.nanni@unipd.it.

methods used in medical image analysis include the scale-invariant feature transform (SIFT) [1], speeded up robust feature (SURF) [2], histogram of oriented gradients (HOG) [3], gradient location and orientation histogram (GLOH) [4], region covariance matrix (RCM) [5], edgelet [6], gray level co-occurrence matrix (GLCM) [7], local binary patterns (LBP) [8], nonbinary encodings [9], color correlogram (CCG) [10], color coherence vectors (CCV) [11], color indexing [12], steerable filters [13] and Gabor filters [14].

A recent method that is simple, effective, and robust and that has consistently produced good results is the Local Binary Pattern (LBP) operator [15]. LBP has been employed in a number of different image analysis methods and has successfully been applied to a large number of medical tasks, including the discrimination of endoscopical images of pediatric celiac diseases [16] and the discrimination of real masses from normal parenchyma in mammographic images [17]. LBP has also proven beneficial in medical data mining, producing excellent results when combined with other descriptors for exploring brain magnetic resonance data [18] and for classifing different cell phenotypes [19].

The main intention of this paper is to compare and combine different approaches for extracting texture descriptors using Dijkstra's algorithm, where each pixel of the image corresponds to a node that is connected to its 4-neighbors. Dijkstra's algorithm is used in the context of graphs for finding the minimum-cost path between two given nodes. It can also be used for extracting information from images if they are mapped onto graphs, an idea recently exploited in [20] to develop a method for texture classification.

In this paper we combine a new set of features proposed here with a previous set proposed in [20]. The descriptors are used to train separate SVM classifiers with results combined by weighted sum rule. We also explore combining our proposed set of features with LBP and a popular LPB variant, local ternary patterns (LTP); when our proposed set of features are coupled with either LBP or LTP, the fusion boosts the performance of our new descriptors. For validating our idea, the performance in six medical datasets representing different image classification problems is reported.

1. Feature Extractors

In this section the feature extraction approaches compared in the experiments are described. In all experiments, SVM [21-23] (using both linear and radial basis function kernels) is the base classifier. For each dataset, the best kernel and the best set of parameters are chosen using a 5-fold cross validation approach on the training data.

1.1. Dijkstra's Algorithm for Texture Descriptors (DJ)

The DJ approach [20] transforms the input image into a graph by connecting each pixel to its 4-neighbors. The cost, *w*, of a transition between two connected nodes at locations (x,y) and (x', y') is determined by their gray levels $I(x,y)$ and $I(x', y')$, such that:

$$w = |I(x,y) - I(x',y')| + \frac{I(x,y) - I(x',y')}{2} \tag{1}$$

Given this model, it is possible to measure the image texture considering the minimum cost of a given set of transitions, namely: i) the central points of the first and last column (P_0), ii) the lower left and upper right corner (P_{45}), iii) the central points of the upper and lower row (P_{90}), and iv) the lower right and upper left corner (P_{135}). The Dijkstra's algorithm [24] is exploited to find the minimum cost path.

It should be observed that the cost w depends on two elements: the difference in the gray levels of the two pixels and their average value. The cost of the four paths (i-iv described above), therefore, depends on the number (and intensity) of the transitions along the four paths, which leads to texture measurement.

As pointed out in [20], the four paths described above can only provide a coarse evaluation of the texture. To achieve more reliable results, the image is subdivided into smaller windows of the same size, and the four paths are evaluated in each subwindow. Several subdivisions of the input image can be obtained by varying the window dimension r and, consequently, the number of windows. In our experimental section, we consider several sets of subwindows, using dimensions based on {10%, 12.5%, 16.6%, 25%, 50%} of the original image.

Texture is measured by means of a descriptor, which is built based on the costs of the four paths $P_0, \dots, P_{135}$ described above so that for each subdivision of the original image into subwindows a vector φ is built by concatenating the average and variance values of the cost of the four paths:

$$\varphi_i = [\mu_0, \mu_{45}, \mu_{90}, \mu_{135}, \sigma_0, \sigma_{45}, \sigma_{90}, \sigma_{135}] \qquad (2)$$

where μ_0 and σ_0 are the mean and variance of all P_0 in the considered i-th subdivision, respectively. The others values of μ and σ refer to the other orientations. The global descriptor is obtained by concatenating all φ_i, one for each subdivision. This global descriptor is finally fed into a classifier to obtain a final decision on the input image.

1.2. New Proposed Features

In this paper we propose a novel method for measuring image texture based on Dijkstra's algorithm. Unlike [20], where the DJ algorithm is used to measure the cost of given paths in the image and in a set of sub-windows, we base our texture measurement on the comparison of the Euclidean distance with the cost calculated by Dijkstra's algorithm. Moreover, we perform such comparisons for all pixels of the image, instead of a small subset, in order to gather a more detailed description.

Our new method needs to define a seed: we chose as the pixel in the top-left corner. The Dijkstra's algorithm is then used to measure the distance of each pixel of the image to the seed. To accomplish this, the image itself is modeled as a graph in which each pixel is connected to its four-neighbors, composed of the adjacent pixels going up, down, left, and right. The cost of each link connecting two pixels depends on their gray levels $G(x,y)$ and $G(x', y')$ and is evaluated as $w=|G_{x,y} - G_{x',y'}|$ if the pixels are adjacent, i.e. if $|x-x'| + |y-y'| = 1$, and infinity otherwise. Note that evaluating the distances of every pixel to the seed does not require a higher computational cost than evaluating the distance between two pixels on opposite sides of an image.

The Dijkstra's costs are then compared with the geometrical distance of the pixels to the same seed. This is possible because the cost of each path depends on two elements: the amount of variations of the gray levels in the pixel matrix (which is related to texture) and the distance of a pixel to the seed. Since we are interested only in the first cost, we remove the second one by calculating the geometrical distance and compensating for its effect. The distance is evaluated considering the L_1-norm, as it reflects the 4-neighbor pixels chosen for evaluating the costs.

Once costs and geometrical distance are available, they undergo a normalization process before being compared. In particular, the maximum value for the L_1-norm is given by $L_{max}=W+H-2$, where W and H are the image width and height, respectively. The maximum value of the Dijkstra cost can be obtained considering the maximum cost for each transition between adjacent pixels: in the case of 8-bit single channel images, the maximum cost for each transition is 255. Given a pixel in the image, the ratio:

$$NDC = \frac{w_{(x,y)}}{L_{1(x,y)}} \quad (3)$$

between its cost and its distance to the seed is forced to be in the range [0, 255]. This ratio is called *NDC* (Normalized Dijkstra Cost). The L_1-norm undergoes a normalization process that remaps it to the same range obtained for NDC: for this reason, the Normalized L_1-Norm (NLN) is calculated as:

$$NLN = \frac{L_{1(x,y)}}{W+H-2} \cdot PR \quad (4)$$

where *PR* (Pixel Range) is the difference between the minimum and maximum gray level (255 in the case of 8-bit images).

Information about costs and distances is organized in a histogram evaluated over a two-dimensional domain. A square matrix of PR rows and PR columns is created: along the x-axis, the NLN is mapped; while along the y-axis, NDC is reported. The image pixels are then scanned, and the histogram bin having coordinates corresponding to NLN and NDC of the considered pixel is incremented by 1. This way of organizing information turns out to be very effective in measuring texture because the bin distribution across the histogram describes how the texture is span over small and larger areas of the image.

For extracting a set of features from the 2D histogram, two methods are used and then combined:

- Method 1, where the discrete cosine transform (DCT) is used to reduce the size of the 2D histogram. The best performance is obtained when DCT is used for extracting a 400 dimensional feature vector.
- Method 2, where only a subset (i.e. the bins where the cumulative value among the training patterns is higher than 10) of the 2D histogram is considered for training SVM.

It should be noted that in our experiments the same configuration is used for all the six datasets.

2. Datasets

Our system was tested on six medical datasets to verify the generality of the proposed approach. In Table 1 a short descriptive summary of each dataset is reported, showing that the chosen datasets represent very different computer vision problems. The datasets used in our experiments are the following:

- PAP (http://labs.fme.aegean.gr/decision/downloads): the pap smear dataset in [25], which contains images representing cells used in cervical cancer diagnosis.
- VIR (http://www.cb.uu.se/~gustaf/virustexture/): the virus dataset in [26], which contains images of viruses extracted by negative stain transmission electron microscopy. We use the 10-fold validation division of images shared by the authors. However, rather than use the masks in [26] for background subtraction, we use the entire image for extracting features, as we found this method produced better results.
- HI (http://www.informed.unal.edu.co/histologyDS): the histopatology dataset in [27], which contains images taken from different organs that represent the four fundamental tissues.
- BC (available upon request from Geraldo Braz Junior [ge.braz@gmail.com]): the breast cancer dataset in [28], which contains 273 malignant and 311 benign breast cancer images.
- PR (available upon request from Loris Nanni, the corresponding author of this paper): the protein dataset in [29], which contains 118 DNA-binding Proteins and 231 Non-DNA-binding proteins, with texture descriptors extracted from the 2D distance matrix that represents each protein. The 2D matrix is obtained from the 3D tertiary structure of a given protein by considering only atoms that belong to the protein backbone (see [29] for details).
- CHO (http://ome.grc.nia.nih.gov/iicbu2008/hela/index.html#cho): the chinese hamster ovary cell dataset in [30], which contains 327 fluorescent microscopy images of ovary cells belonging to five different classes. Images are 16 bit grayscale of size 512×382 pixels.

To reduce the computation time, images larger than 250 pixels are reduced to 250 pixels.

Table 1. Descriptive summary of the six datasets.

Name	Abbreviation	#Classes	#Samples	Sample Size
Histopatology	HI	4	2828	Various
Pap Smear	PAP	2	917	Various
Virus types classification	VIR	15	1500	41 X 41
Breast cancer	BC	2	584	Various
Protein classification	PR	2	349	Various
Chinese Hamster Ovary	CHO	5	327	512 X 382

3. Experimental Results

Tests for each texture descriptor were performed using the 5-fold cross-validation protocol, except for the VR dataset for which we used the original testing protocol described in [26]. Performance was measured considering the area under the ROC curve (AUC) since it provides a better overview of classification results. AUC is a scalar measure that can be interpreted as the probability that the classifier will assign a higher score to a randomly picked positive sample than to a randomly picked negative sample [31]. In the multi-class problem, AUC is calculated using the one-versus-all approach (i.e. a given class is considered as "positive" and all the other classes are considered as "negative") and the average AUC is reported.

The aim of the first experiment, see table 2, is to determine the usefulness of the different techniques applied for extracting features using the paths obtained by Dijkstra's algorithm. In the following, method x×A+y×B is the weighted sum rule between approaches A, with weight x, and B, with weight y. Before fusion, the scores of A and B are normalized to mean 0 and standard deviation 1.

The following approaches are reported in table 2:

- SUB: the method detailed in section 1.1;
- PG-dct: the method based on DCT, detailed in section 1.2;
- PG-sub: the method based on the bins of the 2D histogram, detailed in section 1.2;
- PG: 2×PG-sub + PG-dct;
- DIJ: 2×SUB + PG.

The last column in table 2 labelled *Avg* reports the average performance across the six datasets.

Table 2. Fusion approaches.

	PAP	VIR	CHO	HI	BC	PR	AVG
SUB	79.9	**87.7**	88.7	75.2	77.6	83.4	82.1
PG-dct	76.9	66.4	72.1	60.5	62.3	72.4	68.4
PG-sub	77.2	75.0	63.9	67.6	66.5	69.8	70.0
PG	78.7	75.8	61.3	68.5	65.8	76.0	71.0
DIJ	**81.7**	87.4	**89.9**	**76.6**	**78.6**	**84.9**	**83.2**

Examining table 2, it is clear that our new proposed set of features boosts the performance of SUB, with DIJ outperforming SUB in all datasets except VIR. In this last case, the low dimension of the images lead to a very low number of pixels building the histogram, which in turn did not have a well-defined shape. Shape analysis, therefore, provided poor results since the obtained shape was not informative.

Table 3. Fusion approaches combining LBP and LTP.

	PAP	VIR	CHO	HI	BC	PR	AVG
LBP	87.7	89.8	**99.9**	**92.5**	92.4	79.8	90.3
4xLBP+DIJ	**88.9**	**90.8**	**99.9**	**92.5**	**93.2**	**82.5**	**91.3**
8xLBP+DIJ	88.4	90.4	**99.9**	**92.5**	92.9	81.4	90.9
LTP	86.1	91.6	**99.9**	92.8	**95.6**	87.8	92.3
4xLTP+DIJ	**87.1**	**92.3**	**99.9**	92.7	95.4	**89.5**	**92.8**
8xLTP+DIJ	86.7	92.0	**99.9**	**92.9**	**95.6**	89.0	92.7

In table 3 we show that the proposed set of features can be coupled with LBP and LTP for boosting their performance.

4. Conclusion

In this study we improved the performance of a texture classification method based on Dijkstra's algorithm. We exploited the method for mapping input images into graphs and proposed a new algorithm for defining the transition cost between pixels. We also derived a novel way of mapping Dijstra's costs with respect to geometric distance, building a histogram and analyzing it as a 3D function. We also combined our approach with LBP and LTP, and found that combining LBP and LTP with our novel features boosted their performance.

Our proposed system was validated across six medical image classification problems, with very different images, thereby demonstrating the generality of our approach. For all experiments SVM was used as the base classifier. We also compared our results the state-of-the-art descriptors LBP and LTP. Our study shows that it is worthwhile exploring the idea of mapping images into graphs and exploiting path costs provided by Dijkstra's algorithm to measure image texture.

Future development of this work will include a study involving a feature selection stage for determining which bins are useful and which ones can be neglected: it should be noted that a large number of bins in the histogram are 0 or contain very low values. This was not considered in this work, but could lead to optimization and further considerations of the proposed method.

References

[1] Lowe, D. G., "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision,* vol. 60, no. 2, pp. 91-110, 2004.

[2] Bay, H., Tuytelaars, T., and Gool, L. V., "SURF: Speeded up robust features," *European Conference on Computer Vision,* vol. 1, pp. 404-417, 2006.

[3] Dalal, N., and Triggs, B., "Histograms of oriented gradients for human detection," in 9th European Conference on Computer Vision, San Diego, CA, 2005.

[4] Mikolajczyk, K., and Schmid, C., "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 29, no. 10, pp. 1615-1630, 2005.

[5] Tuzel, O., Porikli, F., and Meer, P., "Region covariance: A fast descriptor for detection and classi?Cation," in 9th European Conference on Computer Vision, 2006, pp. 589-600.

[6] Wu, B., and Nevatia, R., "Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors," in IEEE International Conference on Computer Vision, 2005, pp. 90-97.

[7] Haralick, R. M., Shanmugam, K., and Dinstein, I., "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 3, no. 6, pp. 610-621, 1973.

[8] Ojala, T., Pietikäinen, M., and Harwood, D., "A comparative study of texture measures with classification based on featured distribution," *Pattern Recognition Letters,* vol. 29, no. 1, pp. 51-59, 1996.

[9] Paci, M., Nanni, L., Lathi, A., Aalto-Setälä, K., Hyttinen, J., and Severi, S., "Non-binary coding for texture descriptors in sub-cellular and stem cell image classification," *Current Bioinformatics,* vol. 8, no. 2, pp. 208-219, 2013.

[10]Huang, J., Kumar, S. R., Mitra, M., Zhu, W. J., and Zabih, R., "Image indexing using color correlograms," in IEEE Conference on Computer Vision and Pattern Recognition, San Jose, Puerto Rico, 1997, pp. 762-768.

[11]Pass, G., Zabih, R., and Miller, J., "Comparing images using color coherence vectors," in ACM international conference on Multimedia, 1996, pp. 65-73

[12]Swain, M. J., and Ballard, D. H., "Color indexing," *International Journal of Computer Vision,* vol. 7, pp. 11-32 1991.

[13]Freeman, W. T., and Adelson, E. H., "The design and use of steerable filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 13, pp. 891-906 1991.

[14]Jain, A. K., and Farrokhnia, F., "Unsupervised texture segmentation using gabor filters," *Pattern Recognition Letters,* vol. 24, pp. 1167-1186 1991.

[15]Ojala, T., Pietikainen, M., and Maeenpaa, T., "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 24, no. 7, pp. 971-987, 2002.

[16]Vécsei, A., Amann, G., Hegenbart, S., Liedlgruber, M., and Uhl, A., "Automated marsh-like classification of celiac disease in children using local texture operators," *Computers in Biology and Medicine,* vol. 41, no. 6, pp. 313-25, 2011.

[17]Oliver, A., Lladó, X., Freixenet, J., and Martí, J., "False positive reduction in mammographic mass detection using local binary patterns," *Medical Image Computing and Computer-Assisted Intervention (MICCAI), Lecture Notes in Computer Science 4791*, pp. 286-293, Brisbane, Australia: Springer, 2007.

[18]Unay, D., and Ekin, A., "Intensity versus texture for medical image search and retrieval " in 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 2008, pp. 241-244.

[19]Nanni, L., and Lumini, A., "A reliable method for cell phenotype image classification," *Artificial Intelligence in Medicine,* vol. 43, no. 2, pp. 87-97, 2008.

[20]de Mesquita Sá Junior, J. J., Backes, A. R., and Cortez, P. C., "Texture analysis and classification using shortest paths in graphs," *Pattern Recognition Letters,* vol. 34, no. 11, pp. 1314-1319, 2013.

[21] Vapnik, V., and Lerner, A., "Pattern recognition using generalized portrait method," *Automation and Remote Control,* vol. 24, pp. 774-80, 1963.

[22] Vapnik, V., and Chervonenkis, A., "A note on one class of perceptrons," *Automation and Remote Control,* vol. 25, 1964.

[23] Kuncheva, L. I., and Whitaker, C. J., "Measures of Diversity in Classifier Ensembles and their Relationship with the ensemble accuracy," *Machine Learning,* vol. 51, no. 2, pp. 181-207, 2003.

[24] Dijkstra, E. W., "A note on two problems in connexion with graph," *Numerische Mathematik,* vol. 1, pp. 269-271, 1959.

[25] Jantzen, J., Norup, J., Dounias, G., and Bjerregaard, B., "Pap-smear benchmark data for pattern classification," in Nature inspired Smart Information Systems (NiSIS), Albufeira, Portugal, 2005, pp. 1–9.

[26] Kylberg, G., Uppström, M., and Sintorn, I.-M., "Virus texture analysis using local binary patterns and radial density profiles," in 18th Iberoamerican Congress on Pattern Recognition (CIARP), 2011, pp. 573-580.

[27] Cruz-Roa, A., Caicedo, J. C., and González, F. A., "Visual pattern mining in histology image collections using bag of features," *Artificial Intelligence in Medicine*, no. 52, pp. 91-106, 2011.

[28] Junior, G. B., Cardoso de Paiva, A., Silva, A. C., and Muniz de Oliveira, A. C., "Classification of breast tissues using Moran's index and Geary's coefficient as texture signatures and SVM," *Computers in Biology and Medicine,* vol. 39, no. 12, pp. 1063-1072, 2009.

[29] Nanni, L., Shi, J.-Y., Brahnam, S., and Lumini, A., "Protein classification using texture descriptors extracted from the protein backbone image," *Journal of Theoretical Biology,* vol. 3, no. 7, pp. 1024-1032, 2010.

[30] Boland, M. V., Markey, M. K., and Murphy, R. F., "Automated recognition of patterns characteristic of subcellular structures in fluorescence microscopy images," *Cytopathology,* vol. 33, no. 3, pp. 366-75, 1998.

[31] Fawcett, T., *ROC graphs: Notes and practical considerations for researchers*, HP Laboratories, Palo Alto, USA, 2004.