A Directional Texture Descriptor via 2D Walking Ant Histogram

Serkan Kiranyaz, Miguel Ferreira* and Moncef Gabbouj[1]

*Tampere University of Technology, Tampere, Finland, *University of Twente, Netherlands*

*{serkan.kiranyaz, moncef.gabbouj}@tut.fi, *mpaf98@yahoo.com*

Abstract

A novel texture descriptor, which can be extracted from the major object edges automatically and used for the content-based retrieval in multimedia databases, is presented. The proposed method is adopted from the 2D Walking Ant Histogram, which is in fact a generic shape descriptor recently developed for general purpose multimedia databases. 2D WAH shape descriptor is motivated from the imaginary scenario of a walking ant with a limited line of sight over the boundary of a particular object; eventually each sub-segment is traversed and the process keeps describing a certain line of sight, whether it is a continuous branch or a corner, using individual 2D histograms. In this paper we tuned this approach as an efficient texture descriptor, which achieves a superior performance especially for directional textures. Integrating the whole process as feature extraction module into MUVIS framework allows us to test the mutual performance of the proposed texture descriptor in the context of multimedia indexing and retrieval.

1. Introduction

In the area of content-based multimedia indexing and retrieval, there is a lack of a generic and robust shape descriptor and the existing methods are merely applicable on such databases where object shapes are extracted manually (e.g. binary shape databases). Alternatively, the efforts are mainly focused on edge-based approaches since the edge field in an image usually represents both object boundaries and texture. MPEG-7 Edge Histogram (EHD) [7] generates a histogram of the main edge directions (vertical, horizontal and two diagonals) within fixed size blocks. It is an efficient texture descriptor for the images with heavy textural presence. It can also work as a shape descriptor as long as the edge field contains the true object boundaries and not saturated by the background texture. In this case the method is particularly efficient on describing geometric objects due to its block-based edge representation only with four directions. A similar but pixel-based method applied directly over Canny edge field [2] is called Histogram of Edge Directions (HED) [1]. Another approach, so called Angular Radial Partitioning (ARP), is presented in [4]. ARP basically works over radial blocks (angular slices from quantized radial steps from the center of mass of a re-scaled image). Although rotation invariance can be obtained within this method, the shape outlines are degraded due to the loss of aspect ratio during re-scaling of the image into square dimensions to fit a surrounding circle. A promising method, Edge Pixel Neighborhood Histogram (EPNH) [3], creates a 240-bin histogram from the direction of the neighbor edge pixels. Although it can describe only one-pixel neighborhood over the entire edge field, it exhibits a comparable performance to MPEG-7 EHD. Nevertheless, all these methods turn out to be texture descriptors since they cannot discriminate the true object boundaries that are usually suppressed from the surrounding texture edges.

2D Walking Ant Histogram (2D WAH) [6] is initially developed to address this problem as a generic shape descriptor, which works as long as the majority of object edges are available yet the full object (boundary) extraction may or may not be possible. So the main advantage of it is that it can still describe a shape from its rough sketch with some missing parts. It works over the edge field of the image; however ordinary images are usually too "detailed" to achieve an accurate shape extraction over the edge field. Therefore, as proposed in [5] the relevant *sub-segments*, which are characterized by long, connected series of relatively strong edge-pixels, are extracted from the *scale-map* as the first step and then a novel shape description, as referred to 2D Walking Ant Histogram (WAH), is applied over them. It is basically motivated from the following imaginary scenario; suppose an ant is walking over a solid object and every once in a while, say in a few steps, it "describes" its "Line of Sight (*LoS*)" in a convenient way. It can eventually perform a detailed (high resolution) description since it is quite small compared to the object. So cumulating all the intermediate *LoS* descriptions in a (2D) histogram, particularly focusing on continuous branches and major corners, yields an efficient cue about the shape. Such a description is still feasible if some portion of the object boundary is missing and this is essentially the major advantage of this method. The description frequency (i.e. how often the ant makes a new –intermediate- description) and the length of *LoS* will obviously be the two major parameters of this scheme. The third one is the amount (number) of relevant *sub-segments* that are taken into consideration (description). Keeping this number sufficiently low yields the method to describe only the major object(s') boundaries whilst discarding the texture edges. In this paper, we reverse this process and configure 2D WAH as a texture descriptor by performing necessary manipulations and changes on the generic overview, yet keeping the primary 2D WAH structure intact, that is, extracting the necessary amount of *sub-segments* from the edges of the texture and describing them via (branch) 2D WAH histogram.

The proposed method is fully automatic (i.e. without any supervision, feedback or training involved). Forming the whole process as a *FeX* module into MUVIS framework, [8], allows us to test the overall performance in the context of multimedia indexing and retrieval. Accordingly we will make comparative evaluations through existing edge based texture descriptors (e.g. MPEG-7 EHD, EPNH and ARP) mentioned earlier, as well as generic and powerful texture descriptors such as Gabor [9], Gray Level Co-occurrence Matrix (GLCM) [10], and Ordinal Co-occurrence Matrix (ORDC) [11]. The rest of the paper is organized as follows: Section 2 presents

[1] This work was supported by the Academy of Finland, project No. 213462 (Finnish Centre of Excellence Program (2006 - 2011)

an overview about the proposed method, presenting especially the modifications performed to configure 2D WAH scheme as a texture descriptor. We also discuss the formation of branch 2D WAH from *sub-segments* and implementation of the proposed method as a *FeX* module used for both indexing and retrieval of MUVIS multimedia databases. Experimental results are given in Section 3 and Section 4 concludes the paper.

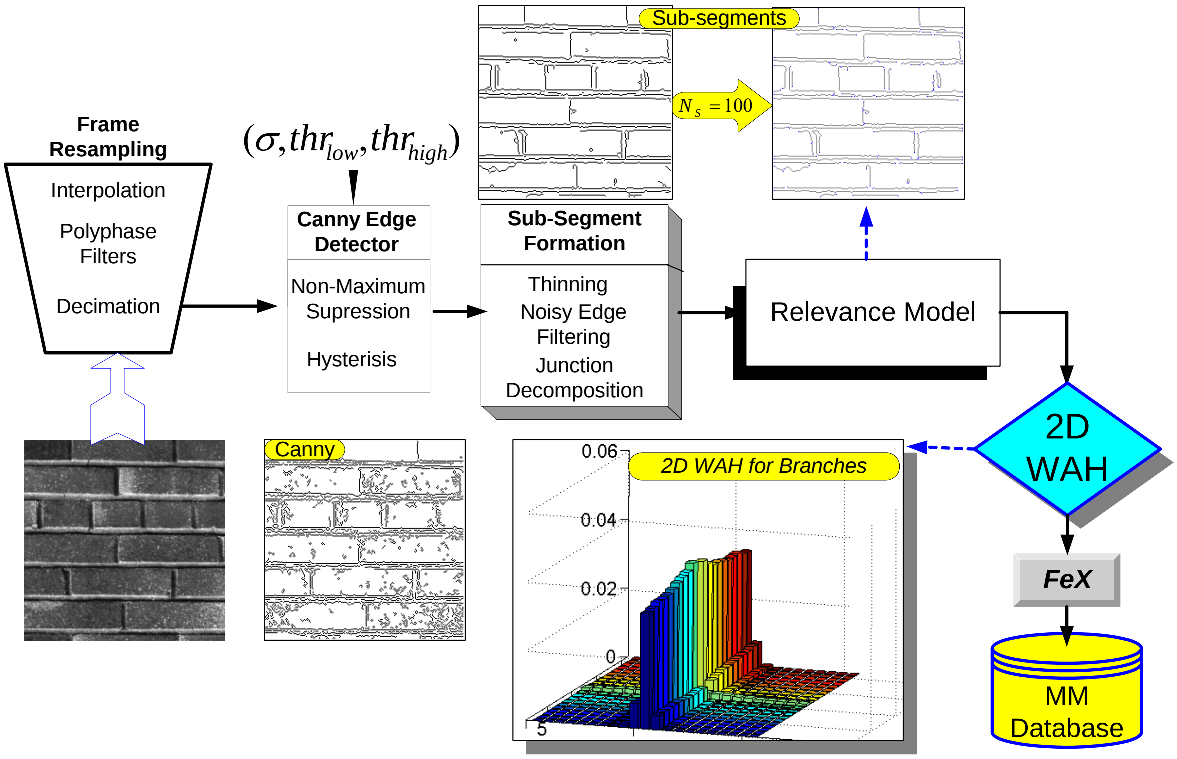

Figure 1: Overview of the 2D WAH as a texture descriptor.

2. Configuration of 2D WAH as a Texture Descriptor

As detailed in [6], 2D WAH as a shape descriptor requires a certain preliminary phase, which mainly consists of four major parts: Frame resampling (size transformation), Bilateral filtering and *scale-map* formation over Canny edge field, *sub-segment* formation and analysis and finally the selection of the relevant *sub-segments* using a relevance model. Once the required number of relevant *sub-segments* is selected, 2D WAH is formed distinctively for both branches and corners. Feature eXtraction (*FeX*) process transforms the 2D WAHs in a suitable descriptor (feature vector) for MUVIS indexation. The iterative Bilateral Filtering [5], [12], is basically needed to remove textural details and to form the scale map, which assigns the maximum scale factor to each edge pixel as long as it prevails and therefore, both (edge) localization and scale information can be obtained from it accordingly. From the scale-map, the *sub-segments* are formed. A *sub-segment* can basically be defined as the series of connected edge pixels with two end-points and they are formed after the following pre-processing steps directly applied onto edge field of the *scale-map*: *One-Pixel Thinning, "Noisy Edge" Removal, Junction Decomposition.* More detailed information about *sub-segment* formation can also be obtained in [5].

As shown in Figure 1, in order to reverse the process of texture removal, Bilateral Filtering is entirely left out via assigning zero to number of scales and henceforth, the *sub-segments* are extracted directly over the initial edge field. There is still the need of *sub-segment* formation, since they represent the connected series of edge pixels after the aforementioned pre-processing stage. Especially making them one-pixel thick avoids the redundant edge representation on 2D WAH from which the edge directional information can be accurately obtained. Since the scale information is not needed, the Relevance Model uses only the length of the *sub-segments*, so as to select the longest N_S *sub-segments*. 2D WAH as a shape descriptor N_S can initially be set to a sufficiently large number in order not to loose relevant *sub-segments;* however, it should not too large to avoid significant disturbance of irrelevant (textural) segments (i.e. $10 \le N_S < 25$). Therefore, it can be set large enough (i.e. $N_S \ge 100$) to transform 2D WAH to a texture descriptor respectively so that the majority of the edges can now be considered for 2D WAH formation.

The longest N_S *sub-segments,* whether in closed loop (CL) or non-closed loop (NCL) form are used for the formation of branch 2D WAH. In order to accomplish this, for instance on a NCL *sub-segment*, the ant with a certain *LoS* (say L_S pixels long) and a walking pace Δ (i.e. $\Delta < L_S$), traverses from one end-point to the other, whilst describing each *LoS* section into branch WAH, via incrementing the corresponding 2D bins, which actually represent the pixel grid centred from the first and last pixels of *LoS*. For a CL sub-segment, traversing can start from any point and terminates whenever it looped-back to the start. Another major difference between the shape and texture configurations is the role of corner detection. The corners bears an important information for describing a particular shape; however, they are not too relevant for textures as they might have arbitrary orientation and angle even for similar textures. Therefore, the corner WAH is not formed in the proposed texture configuration and only branch WAH is used for description.

Let p_1 and p_2 be the first and the last pixels of a *LoS* respectively and the number of pixels in between is the length of *LoS* (i.e. $N_p(p_1 \to p_2) = N_p(p_2 \to p_1) = L_S$). In order to achieve an invariant representation with respect to traversing (ant's walking) direction, both *LoS* sections (i.e. $\{\forall p : p \in (p_1 \to p_2)\} \cup \{\forall p : p \in (p_2 \to p_1)\}$) are equally represented in 2D WAH, and therefore it has a dimension of $2L_S - 1 \times 2L_S - 1$. Figure 2 illustrates a sample CL segment and a branch *LoS* representation over the 2D WAH. The walking (traversing) direction is clock-wise and note that it does not make any difference on 2D WAHs since both directions are considered.

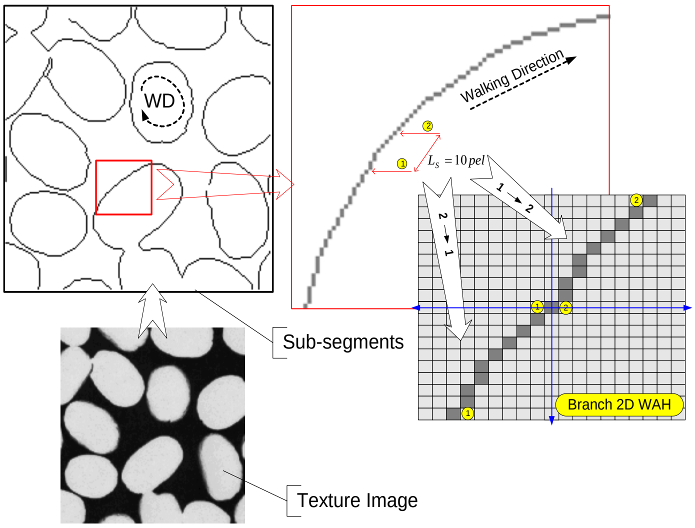

Figure 2: Localization of a branch *LoS* into 2D WAH.

So all *LoS* sections, which are encountered within a *sub-segment* (i.e. $\{\forall p : p \in (p_1 + n\Delta \to p_2 + n\Delta)\},\ n = 0,1,2,...$) can be cumulated in the branch WAH. As shown on the right side of Figure 2, the corresponding bins of branch WAH (for both *LoS* localization) are simply incremented by one. The process continues with the next *LoS* section (proceeds Δ pixels towards the walking direction) and keeps moving until encountering the end of the *sub-segment* within the *LoS* section. Once the traversing over a CL or NCL sub-segment is completed, then the same process is repeated for the next one, visiting all the N_S *sub-segments* selected. Due to space limitations, the conversion of 2D WAH into a feature vector, for MUVIS database indexation is skipped in this paper.

The retrieval process in MUVIS is based on the traditional query by example (QBE) operation. The features of the query item are used for (dis-) similarity measurement among all the features of the visual items in the database. Ranking the database items according to their similarity distances yields the retrieval result. Let $\overline{WAH}^{B,q}_{i,j}$, $\overline{WAH}^{B,x}_{i,j}$ be the (unit-normalized) branch WAHs of the query (*q*) and a particular database item (*x*). The total (dis-) similarity distance, $D(q,x)$, computed directly from the branch WAHs can be expressed as follows.

$$D(q,x) = \frac{1}{2} \sum_{i=-L_S+1}^{L_S-1} \sum_{j=-L_S+1}^{L_S-1} \left| \overline{WAH}^{B,q}_{i,j} - \overline{WAH}^{B,x}_{i,j} \right| \qquad (1)$$

Due to unit normalization of the particular branch WAHs during the indexing process, $D(q,x) \le 1$ as required by MUVIS retrieval scheme [8].

3. Experimental Results

All texture-based retrievals are evaluated using the *ground-truth* methodology whilst providing both visual and numerical results. In the experiments performed in this section, we used **Texture** Image database with 1760 texture images that are obtained from *Brodatz* database. A query image is chosen among the database items to be the "Example" and a particular *FeX* module (e.g. ARP) is selected to retrieve and rank the similar (based on shape) images using only the respective (ARP) features and an appropriate distance metric implemented within the *FeX* module. The recommended distance metrics are implemented for each *FeX* module, i.e. Euclidean (L_2 norm) for MPEG7 EHD, Manhattan (L_1 norm) for ARP and weighted city-block for EPNH, respectively. For MPEG7 EHD we set the block size to 8. 201x201 is the normalized frame size, 12 and 8 are angular and radial partition numbers for ARP, respectively. For 2D WAH, we set $L_S = 10$, $\Delta = 1$ and $N_S = 100$.

In order to evaluate the texture-based retrieval performance, we run the proposed method along with the other methods over **Texture** database generated from **Brodatz** texture collection. In order to measure the retrieval performance, we used an unbiased and limited formulation of *Normalized Modified Retrieval Rank* (*NMRR(q)*), which is defined in MPEG-7 as the retrieval performance criteria per query (*q*). It combines both of the traditional hit-miss counters; *Precision – Recall,* and further takes the ranking information into account as given in the following expression:

$$\begin{aligned} AVR(q) &= \frac{\sum_{k=1}^{N(q)} R(k)}{N(q)} \quad and \quad W = 2N(q) \\ NMRR(q) &= \frac{2AVR(q) - N(q) - 1}{2W - N(q) + 1} \le 1 \\ ANMRR &= \frac{\sum_{q=1}^{Q} NMRR(q)}{Q} \le 1 \end{aligned} \qquad (2)$$

where *N(q)* is the minimum number of relevant (via *ground-truth*) images in a set of *Q* retrieval experiments, *R(k)* is the rank of the *k*th relevant retrieval within a window of *W* retrievals, which are taken into consideration during per query, *q*. If there are less than *N(q)* relevant retrievals among *W* then a rank of *W*+1 is assigned for the remaining (missing) ones. *AVR(q)* is the average rank obtained from the query, *q*. Since each query item is selected within the database, the first retrieval will always be the item queried and this obviously yields a biased *NMRR(q)* calculation and it is, therefore, excluded from ranking. Hence the first relevant retrieval (*R(1)*) is ranked by counting the number of irrelevant images *a priori* and note that if all *N(q)* retrievals are relevant, then *NMRR(q)*=0, the best retrieval performance is thus achieved. On the other hand, if none of relevant items can be retrieved among *W* then *NMRR(q)*=1, as the worst case. Therefore, the lower *NMRR(q)* is the better (more relevant) the retrieval is, for the query, *q.* Keeping the number of QBE experiments sufficiently high, the average *NMRR*, *ANMRR*, as expressed in Eq. (2) can thus be used as the retrieval performance criteria.

Figure 3 presents *NMRR* plots with the query images using the following parameters: *N(q)*=11, *W*=22 and *Q*=20. Note that 2D WAH achieves the best performance for all queries without any exception and the *ANMRR* results presented in Table I furhter approve that superior (texture-based) retrieval accuracy is obtained.

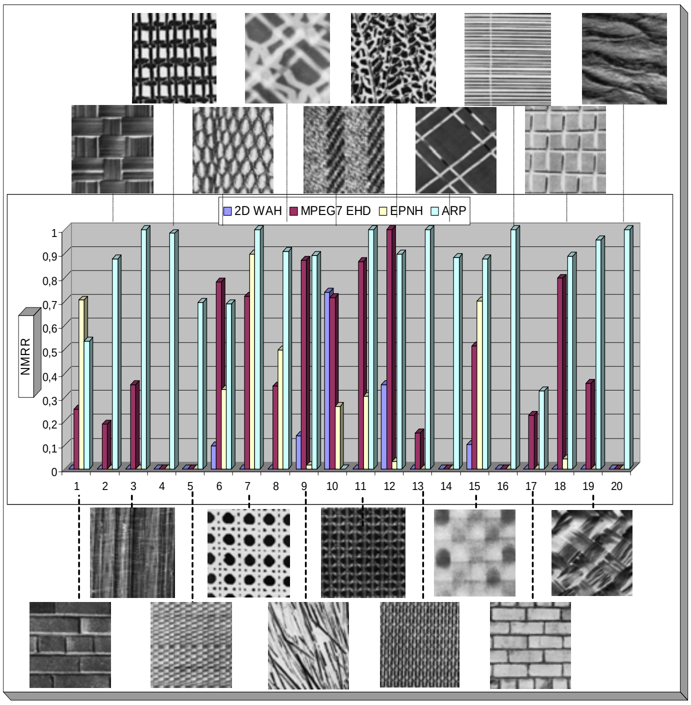

Figure 3: *NMRR* plots of 4 methods for 20 queries over Texture database.

Table I: *ANMRRs* of 4 edge-based methods.

Database	2D WAH	MPEG7 EHD	EPNH	ARP
Texture	0.034	0.407	0.19	0.82

Finally, we also compared the proposed method against the well-known texture descriptors, such as Gabor [9] with 3 scales and 4 orientations, Gray Level Co-occurrence Matrix (GLCM) [10] with block size 8 and Ordinal Co-occurrence Matrix (ORDC) [11]. The results presented in Table II indicate that equal or better retrieval performance is achieved for the directional textures, as shown in Figure 3.

Table II: *ANMRRs* of 4 generic texture descriptors.

Database	2D WAH	Gabor	GLCM	ORDC
Texture	0.034	0.037	0.285	0.085

4. Conclusion

The 2D WAH descriptor presented in this paper is a multi-purpose descriptor which can be configured as a shape or texture descriptor for general-purpose multimedia databases. In this paper we presented its configuration as a texture descriptor and compared its retrieval performance with the traditional methods. The experimental results approve that it can achieve superior retrieval accuracy especially for directional textures. Moreover, its texture-based retrieval performance is equal or better than well known texture descriptors such as Gabor, ORDC and GLCM. The proposed descriptor is computationally efficient since all the time consuming operations in 2D WAH as a shape descriptor, such as iterative Bilateral Filtering, scale-map formation, corner detection, etc. are removed and it is approximately 12 times faster than Gabor texture extraction with the typical settings. Finally as opposed to some other generic descriptors such as Gabor, whenever describing the ordinary images, it has a natural advantage to be performed only the over the regions with real texture presence since it is based on the edge information and thus the regions without any textural structure (e.g. homogenous and smooth areas) are automatically discarded. In this way a pure and unbiased texture description can be obtained.

References

[1] M. Abdel-Mottaleb, "Image retrieval based on edge representation", in *Proc. 2000 International Conference on Image Processing*, Piscatway, NJ, USA, 2000, vol. 3, pp. 734–737.

[2] J. Canny, "A Computational Approach to Edge Detection", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679-698, November 1986.

[3] A. Chalechale and A. Mertins, "An abstract image representation based on edge pixel neighbor-hood information (EPNI)," *Lect. Notes Comput. Science,* 2510, pp. 67-74, 2002.

[4] A. Chalechale, A. Mertins, G. Naghdy ,"Edge Image Description Using Angular Radial Partitioning " *IEE Proc. Vis. Image Signal Proc.,* Vol 151, No. 2, April 2005.

[5] M. Ferreira, S. Kiranyaz and M. Gabbouj, "Multi-scale Edge Detection and Object Extraction for Image Retrieval", ICASSP 2006, Toulouse, France, May 15-19, 2006.

[6] M. Ferreira, S. Kiranyaz and M. Gabbouj, "A Novel Shape Descriptor over Multi-Scale Edge Field: 2D Walking Ant Histogram", in Proc. of IWSSIP 2006, pp. 475-378, Budapest, Hungary, Sep. 21-23, 2006

[7] B. S. Manjunath, J. R. Ohm, V. V. Vasudevan, A. Yamada, "Color and Texture Descriptors", *IEEE Transactions on Circuits and Systems for Video Technology,* Vol 11, No. 6, pp. 716-719, 2001.

[8] MUVIS. http://muvis.cs.tut.fi

[9] W. Y. Ma, B. S. Manjunath,"A Comparison of Wavelet Transform Features for Texture Image Annotation", *Proc. IEEE International Conf. On Image Processing*, 1995.

[10] M. Partio, B. Cramariuc, M. Gabbouj, A. Visa, "Rock Texture Retrieval Using Gray Level Co-occurrence Matrix", *Proc. of 5th Nordic Signal Processing Symposium*, Oct. 2002.

[11] M. Partio, B. Cramariuc and M. Gabbouj, "Block-Based Ordinal Co-occurrence Matrices for Texture Similarity Evaluation," *Proc. 2005 IEEE International Conf. on Image Proc.*, vol. I, pp. 517-520, Genova, Italy, 11-14 Sep. 2005.

[12] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images*", In Proc. of the Sixth International Conference on Computer Vision,* Bombay, India, January 1998.