Multi-Dimensional Dynamic Time Warping for Image Texture Similarity

Rodrigo Fernandes de Mello[1] and Iker Gondra[2]

[1] University of São Paulo
Institute of Mathematics and Computer Science
São Carlos, SP, Brazil
mello@icmc.usp.br
[2] St. Francis Xavier University
Department of Mathematics, Statistics and Computer Science
Antigonish, NS, Canada
igondra@stfx.ca

Abstract. Modern content-based image retrieval systems use different features to represent properties (e.g., color, shape, texture) of the visual content of an image. Retrieval is performed by example where a query image is given as input and an appropriate metric is used to find the best matches in the corresponding feature space. Both selecting the features and the distance metric continue to be active areas of research. In this paper, we propose a new approach, based on the recently proposed Multidimensional Dynamic Time Warping (MD-DTW) distance [1], for assessing the texture similarity of images with structured textures. The MD-DTW allows the detection and comparison of arbitrarily shifted patterns between multi-dimensional series, such as those found in structured textures. Chaos theory tools are used as a preprocessing step to uncover and characterize regularities in structured textures. The main advantage of the proposed approach is that explicit selection and extraction of texture features is not required (i.e., similarity comparisons are performed directly on the raw pixel data alone). The method proposed in this preliminary investigation is shown to be valid by proving that it creates a statistically significant image texture similarity measure.

Keywords: Content-Based Image Retrieval, Texture, Dynamic Time Warping, Similarity Measure, Distance Measure, Chaos Theory.

1 Introduction

In recent years, the rapid development of information technologies and the advent of the Web have accelerated the growth of digital media and, in particular, image collections. As a result, new mechanisms to search on large image databases have been proposed. One of the first approaches was keyword-matching, which uses a textual representation and is based on the manual annotation of images with descriptive keywords. This approach is not only subjective and error-prone but also very time-consuming and cumbersome for large databases.

G. Zaverucha and A. Loureiro da Costa (Eds.): SBIA 2008, LNAI 5249, pp. 23–32, 2008.

Recently, automatic image labeling approaches [2,3,4] have been proposed as an attempt to improve the manual annotation of images. In [2], image recognition techniques are employed to automatically assign a limited number of descriptive keywords. This approach is limited by the fact that current image recognition methods are not completely reliable and, as a consequence, the assigned keywords have to be verified by a person. Other works such as [3] consider the textual context of images, in web pages, to automatically extract descriptive keywords (such as those that appear in captions). The performance of those approaches is lower than the one obtained by using manual annotation. Furthermore, their applicability is limited in situations where there is no textual context (such as in photo albums). Those textual description approaches can only obtain part of the richness and complexity of an image's visual content.

To overcome these problems, content-based image retrieval (CBIR) [5] was proposed in the early 1990's. The basic idea is to directly use the visual content when determining image similarity. Retrieval is performed by using a query image as input, which has a feature set extracted to represent its visual content (e.g., color, shape, texture). Afterwards, an appropriate metric is applied to find the best matches in the corresponding feature-set space. In this context, texture is one of the most important visual characteristics when defining similarity among images.

Texture is defined as the repetition of a certain atomic pattern (or texton) residing in a region. For a low-level image analysis, texture features play a very important role in distinguishing textured regions from one another based on the measurement of optical homogeneity of surfaces. In the 1970's, Haralick et al. [6] proposed the first systematic analysis of texture by using a co-occurrence matrix, which describes the distribution of co-occurring pixel values at a given offset. Based on such matrix, several texture features (e.g., contrast, entropy), which explore the spatial dependence of pixel values, can be extracted. Other researchers carried on investigating texture. For example, Tamura et al. [7] developed a set of texture features designed to measure the visual properties of coarseness, contrast, directionality, line-likeness, regularity, and roughness which, based on conducted psychological experiments, are thought to dominate human visual perception of texture. Research into other techniques, such as the use of wavelet-based texture features, has also been very active.

When considering texture similarity, the selection of both a set of features and a distance metric continues to be the most critical decision. The selection of features requires the application of methods to extract the most relevant visual characteristics, which are then used to compare to other textures. The distance metric is responsible for the comparison of the feature values of different textures.

The complexity of feature extraction and the observation that structured texture contains regular repeated patterns has motivated this work to investigate the suitability of the Dynamic Time Warping (DTW) distance [8] as a feature-independent (i.e., based on the raw pixel values only) measure of image texture similarity. DTW was designed to find the minimal distance between two series considering their synchronization through shifts. In this context, we consider raw pixel data to compute texture similarity. All the relevant features of a texture are, in some way, "hidden" in the raw information. Besides that, any feature extraction technique has its limitations because, as of yet, no one has completely figured out and implemented a technique that is able to obtain the same characterization quality as a person. Thus, instead of attempting to determine texture similarity based on a small set of (probably incomplete) features, we find that it may be advantageous to use a similarity measure that is based on the raw data itself.

The objective of this research is to obtain some preliminary evidence as to whether a DTW-based feature-independent texture similarity measure can actually result in a good, and statistically significant, retrieval performance. This paper is organized as follows: Section 2 introduces the chaos theory concepts of Embedded and Separation dimensions, which are used as a preprocessing step to obtain a descriptive representation of textures. In Section 3 we introduce the traditional and multi-dimensional DTW. Our approach (and its evolution) is described in section 4. Experimental results with a real data set are presented in Section 5. Finally, concluding remarks are given in Section 6.

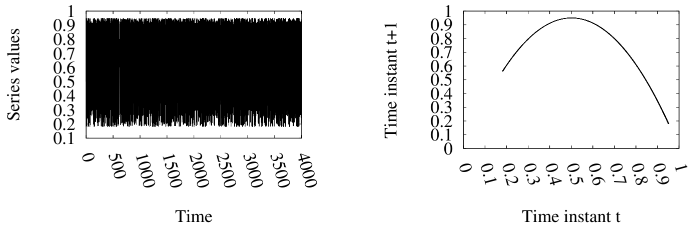
2 Embedded and Separation Dimensions

The occurrence of repeated patterns, in structured textures, has encouraged us to investigate a similarity measure which considers textures as series of events. From that, we decided to consider DTW, a technique that can measure the minimal distance between two series, by considering possible synchronization points. Although, during our studies, we observed that the original image representation may not be the most suitable when uncovering the regularities and patterns of textures. Based on such conclusion, we decided to consider chaos theory tools to unfold and reorganize data textures according to possible hidden regularities. The concepts considered in this work are presented next.

Chaos theory is defined as the qualitative study of unstable aperiodic behavior in deterministic nonlinear dynamical systems [9,10]. Attempts to understand the behavior of such systems have led to the development of several works (e.g., [10,11]) which aim at characterizing possible internal regularities. In order to understand how the internal regularities can be obtained, consider the logistic equation (1) with the initial conditions $t \in [0, 4000]$, $b = 3.8$ and $x_0 = 0.5$.

$$x_{t+1} = b \times x_t \times (1.0 - x_t) \tag{1}$$

Figure 1(a) presents the outputs for the logistic function. If we directly apply a similarity measure comparing this series to another one, we may obtain bad results. A better solution comes by applying chaos theory tools to unfold the full series behavior through two dimensional analysis: the embedded and the separation dimensions. By obtaining those two dimensions, we can understand how information has to be represented in the space, before using it in further comparisons (in the case of texture retrieval).



(a) Outputs (b) Reconstructed with $m = 2$, $\tau = 1$

Fig. 1. Logistic Function

Whitney [12] proposed that a series $x_0, x_1, ..., x_{n-1}$ could be reconstructed into a multidimensional space $x_n(m, \tau) = (x_n, x_{n+\tau}, ..., x_{n+(m-1)\tau})$ where m is the embedded dimension and τ is a fixed time delay. According to this study, each series can be reconstructed and, consequently, simplified to be understood and compared. To better understand those dimensions, consider the same logistic function outputs reconstructed in a multidimensional space where $m = 2$ and $\tau = 1$. This reconstruction, which is basically the plot of x_t versus x_{t+1}, is presented in figure 1(b). Now the behavior of the logistic function, which was apparently a random walk (figure 1(a)), can be understood and modeled in an easier way.

Basically, the embedded dimension defines the number of axis that we will plot the series to unfold its full behavior. Some series can only be understood when using more than two dimensions. Besides the embedded, there is the separation dimension which helps to extract the periodic behavior of a series. This basically tells the number of points we may look back in the history to detect regular behavior (or patterns, this is also known as the seasonability of the series). Next we discuss how to determine good values for both dimensions.

According to Abarbanel [13] we have to apply the autocorrelation function (equation (2), where $E[.]$ is the expected value, μ is the average, k is the time shift size and σ^2 is the variance) on a series and use its first minimum as the separation dimension. The autocorrelation measures how well the series matches itself considering a time separation. This is useful to find repeated patterns in a series. However, this technique is formulated for linear series, and, consequently, it may not present good results for non-linear and chaotic ones.

$$ACF(k) = \frac{E[(X_i - \mu)(X_{i+k} - \mu)]}{\sigma^2} \tag{2}$$

Fraser and Swinney [14] studied and confirmed that the Auto Mutual Information (AMI) technique presents better results to estimate the separation dimension. This technique is not linear-dependent, and consequently is more interesting for this work. To use this technique we must calculate it for different time shifts and adopt the first minimum of the function. The average mutual information is defined in equation (3) where X and Y have, respectively, the probability distribution functions P_X, P_Y and X and Y occur in pairs with the joint distribution P_{XY} [15].

$$I(X;Y) = \int dxdy P_{XY}(x,y) \log_2 \frac{P_{XY}(x,y)}{P_X(x)P_Y(y)} \tag{3}$$

After defining the separation dimension we must find the embedded dimension. Takens [16] and Mañé [17] studied and confirmed that the upper limit of the embedded dimension D_e, an integer value, can be defined using the fractal dimension D_f according to $D_e > 2.0 \times D_f$. Although this dimension is usually larger than necessary. For instance, the fractal dimension of the Lorenz attractor is 2.06 [18], consequently the upper limit of the embedded dimension would be $D_e > 2.0 \times 2.06$, then it is 5. Although, according to [19] the attractor can be represented in $D_e = 3$. The important point here is that we can represent the attractor with a number of dimensions smaller than 5, which reduces complexity. From a mathematical point of view, using either 3 or 5 results in no difference, because once the attractor is unfolded the analysis can be conducted. Although, if we unnecessarily work with more dimensions, we add complexity and execution time to obtain solutions [19]. This conclusion motivated works on how to define a good embedded dimension for different series.

The traditional approach to obtain the minimum embedding dimension is by calculating any system invariant (such as the Lyapunov exponent) to different embedded dimension values and observe when it saturates. The complexity of this approach motivated Kennel *et al.* [19] to propose the False Nearest Neighbors (FNN) method. In FNN, the nearest neighbors for each point, in the space, are calculated, initially with the embedded dimension equal to 1. Then, the Euclidean distance from the point to its nearest neighbor is calculated. Afterwards, a new dimension is added and the distance of the point to its nearest neighbor obtained. If this distance increases, those two points are considered false neighbors. This happens because the attractor being modeled needs more dimensions to be unfolded and studied.

Kennel *et al.* [19] consider a embedded dimension d where the r^{th} nearest neighbor of $y(n)$ is $y^{(r)}(n)$. The Euclidean distance between the point $y(n)$ and its r^{th} nearest neighbor is obtained by equation (4). Adding a new dimension, we go for $d+1$ and add the coordinate $(d+1)^{th}$ in each vector $y(n)$. The new coordinate is $x(n+Td)$ which is included in the new Euclidean distance equation (5).

$$R_d^2(n,r) = \sum_{k=0}^{d-1} (x(n+kT) - x^{(r)}(n+kT))^2 \tag{4}$$

$$R_{d+1}^2(n,r) = R_d^2(n,r) + (x(n+dT) - x^{(r)}(n+dT))^2 \tag{5}$$

Then, the criterion is to measure the distance variation when adding the new dimension as presented in equation (6).

$$V_{n,r} = \sqrt{\frac{R_{d+1}^2(n,r) - R_d^2(n,r)}{R_d^2(n,r)}} = \frac{|x(n+Td) - x^{(n)}(n+Td)|}{R_d^2(n,r)} \tag{6}$$

The authors indicate that, if $V_{n,r} > R_{tol}$, then the points are considered false neighbors, where R_{tol} is a threshold. They conclude that $R_{tol} \geq 10.0$ is enough to generate good results. This reconstruction, using the embedded and separation dimensions, unfolds the attractor and can be applied to any series. After unfolding, we can better and more easily study the behavior of a series. Then, we use the new dataset (the reorganized representation of the image) to compare against others. This new representation has less complexity to understand the image regularities and, consequently, to model it.

3 Multi-Dimensional Dynamic Time Warping

The distance between two hypothetical series can be quantified using different measures, one of them is Dynamic Time Warping (DTW) [8]. This technique aligns two series to find the ideal warp (the best synchronization point) in order to minimize the distance between them.

In order to understand it, consider two series $S = s_0, s_1, ..., s_{m-1}$ and $T = t_0, t_1, ..., t_{n-1}$ of length m and n, respectively. Firstly, DTW (algorithm 1) creates an m-by-n matrix d where each element (ith, jth) represents the distance $d(S_i, T_j) = (S_i - T_j)^2$ between each pair of points S_i and T_j. Afterwards, DTW creates a new matrix D to accumulate the total distance between each possible pair of points of the two series. This step fills out the matrix D where the elements represent all possible alignments (synchronizations) of the two series and their distances.

After calculating the matrix, the DTW distance is computed through the summing of the shortest possible path that starts at the right bottom of the matrix and goes up to the left-top element. This path represents the best synchronization between the two series and the sum of all of its matrix elements is the DTW distance. DTW was designed for one-dimensional series. However, there are many applications in which calculating an optimal alignment requires the use of multi-dimensional series. Holt *et al.* [1] proposed the Multi-Dimensional Dynamic Time Warping (MD-DTW), an approach to calculate the DTW by synchronizing multi-dimensional series, which is basically an extension of the original DTW, where the matrix D is created by computing the distance between k-dimensional points (where, differently from the original approach, k can be larger than 1). This approach preprocesses the multi-dimensional series, which must have the same number of dimensions, according to algorithm 2. The last step of this algorithm is the execution of the traditional DTW (algorithm 1) considering the matrix D as the result of the preprocessing phase.

4 Proposed Approach

The first investigations we conducted, using DTW as a similarity measure to retrieve images, generated better results than random retrieval. Firstly, each

Algorithm 1. Dynamic Time Warping Algorithm

Let m and n be the length of the series S and T, respectively.
Let d be the matrix which computes the distance of pairs of values of S and T.
for $i = 0$ to $m - 1$ **do**
  for $j = 0$ to $n - 1$ **do**
    $d[i][j] = (S[i] - T[j])^2$;
  end for
end for
Let D be the matrix with the DTW distance among pairs of elements of series S and T.
$D[0][0] = d[0][0]$;
for $i = 1$ to $m - 1$ **do**
  $D[i][0] = d[i][0] + D[i-1][0]$;
end for
for $j = 1$ to $n - 1$ **do**
  $D[0][j] = d[0][j] + D[0][j-1]$;
end for
for $i = 1$ to $m - 1$ **do**
  for $j = 1$ to $n - 1$ **do**
    $D[i][j] = min(D[i-1][j], D[i-1][j-1], D[i][j-1]) + d[i][j]$;
  end for
end for
The total DTW distance between the two series is stored at the matrix element $D[m-1][n-1]$.

Algorithm 2. Multi-dimensional Dynamic Time Warping Algorithm

Let S, T be two series of dimension K and length n and m, respectively.
Normalize each dimension of S and T separately to a zero mean and unit variance.
Fill the matrix D according to:
$D(i,j) = \sum_{k=1}^{K} |S(i,k) - T(j,k)|$
Consider the matrix D to compute the traditional DTW algorithm (instead of the matrix D of the traditional approach).

RGB color image was converted to the corresponding grayscale image. Afterwards, we applied a Laplace edge detection algorithm to the grayscale images. Then, before applying DTW as the similarity measure, we organized the pixel values of the preprocessed images in two different ways. To better understand them, let $P = \{p_{0,0}, p_{0,1}, ..., p_{1,0}, p_{1,1}, ..., p_{r-1,c-1}\}$, where $0 \le p_{i,j} \le 255$ is the grayscale value of the pixel at the intersection of the ith row and jth column of an image with r rows and c columns, be the matrix representation of an image. In the first way, we organized the series using the pixels in the following order $\{p_{0,0}, p_{0,1}, ..., p_{1,0}, p_{1,1}, ..., p_{r-1,c-1}\}$; in the second approach, the pixels were organized as $\{p_{0,0}, p_{1,0}, ..., p_{0,1}, p_{1,1}, ..., p_{r-1,c-1}\}$.

After organizing the pixels, we conducted experiments comparing images by using the two different orders but the results were inconsistent (i.e., most query images resulted in very different retrieval performances when using the two orders). Thus, the first conclusion was that neither one of this one-dimensional orders was a good representation of image textures. That is, the original texture representation may not be the most suitable for uncovering existing regularities or patterns. This motivated us to study and apply chaos theory tools to perform optimal data reorganization.

For each image, the embedded and the separation dimensions, were first computed. Then, each image was reconstructed into a multidimensional space $x_n(m, \tau) = (x_n, x_{n+\tau}, ..., x_{n+(m-1)\tau})$ where m is the embedded dimension and

τ is the separation dimension. The resulting multi-dimensional series were then used as the input for MD-DTW-based similarity comparisons.

5 Experimental Results

Consider a database consisting of a set of images $\mathcal{D}$. Let x be a query image and $\mathcal{A} \subset \mathcal{D}$ be the subset of images in $\mathcal{D}$ that are relevant to x. After processing x, the image retrieval method generates $\mathcal{R} \subset \mathcal{D}$ as the retrieval set. Then, $\mathcal{R}^+ = \mathcal{R} \cap \mathcal{A}$ is the set of relevant images to x that appear in $\mathcal{R}$. Users want the database images to be ranked according to their relevance to x and then be presented with only the k most relevant images so that $|\mathcal{R}| = k < |\mathcal{D}|$. Thus, images are ranked by their distance to the query image and, in order to account for the quality of image rankings, precision at a cut-off point (e.g., k) is commonly used. Thus, the performance of the image retrieval method is commonly measured by *precision*, which quantifies the ability to retrieve only relevant images and is defined as $precision := \frac{|\mathcal{R}^+|}{|\mathcal{R}|}$.

The objective of our experiments was to obtain preliminary evidence as to whether the proposed approach actually creates a statistically significant image texture similarity measurement. Therefore, we tested its performance against a uniform random retrieval to select the images in $\mathcal{R}$. The *Texture* data set, obtained from MIT Media Lab [20] was used for evaluation. There are 40 different texture images that are manually classified into 15 classes. Each of those images is then cut into 16 non-overlapping images of size 128x128. Thus, there are 640 images in the database. Sample images are shown in figure 2.

Fig. 2. Sample images from *Texture* data set

Each image was used as a query and the precision and recall of a retrieval set of $k \in [1, 640]$ nearest images was measured. The results of our approach and the random retrieval are presented in figure 3, which shows the degradation of precision as k increases. That is, attempting to increase recall results in the introduction of more non-relevant images into $\mathcal{R}$. Thus, precision-recall graphs have a classical concave shape. In order to increase both precision and recall, the curve should move up and to the right (as observed after the initial decrease (i.e., at the middle) of the curve in figure 3) so that both recall and precision are higher at every point along the curve.

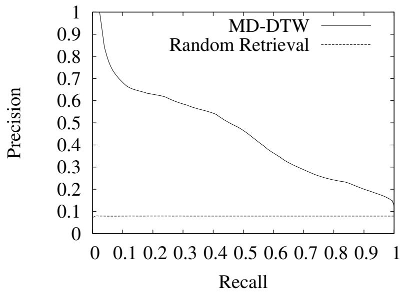


Fig. 3. Precision and recall

It is common to compare retrieval methods by using a fixed retrieval set size. A reasonable (and commonly used) value for the size of the retrieval set $k = 20$. The average precision over the 640 queries with $k = 20$ for the proposed approach and for random retrieval was 0.594 and 0.079 respectively. As can be observed by these results and figure 3, the proposed approach performed surprisingly well and is obviously statistically different than uniform random retrieval. Table 1 shows the average precision for each of the texture classes shown in Figure 2.

Table 1. Average Precision for each of the 15 texture classes shown in figure 2

Class	Average Precision	Class	Average Precision	Class	Average Precision
1	0.974	2	0.970	3	0.797
4	0.998	5	0.422	6	1
7	0.134	8	0.573	9	0.786
10	0.263	11	0.190	12	0.472
13	0.238	14	0.194	15	0.979

6 Conclusions

Based on the observation that structured image textures contain repeated patterns, we investigated the possibility of using the MD-DTW [1], which allows for the comparison of arbitrarily shifted patterns in multi-dimensional series, as a measure of image texture similarity. Chaos theory tools were used in the preprocessing step in order to allow the identification, characterization and unfolding of regularities in the raw data of structured textures. The main advantage of the proposed approach is that explicit selection and extraction of texture features is not required (i.e., similarity comparisons are performed directly on the raw pixel data alone). The proposed approach performed surprisingly well when compared against uniform random retrieval thus proving that it creates a statistically significant image texture similarity measure. This is an encouraging preliminary result that motivates us to continue to work on the MD-DTW as a feature-independent measure of texture similarity.

References

1. ten Holt, G.A., Reinders, M.J.T., Hendriks, E.A.: Multi-Dimensional Dynamic Time Warping for Gesture Recognition. In: Annual Conference of the Advanced School for Computing and Imaging (2007)
2. Ono, A., Amano, M., Hakaridani, M., Satoh, T., Sakauchi, M.: A flexible content-based image retrieval system with combined scene description keywords. In: IEEE International Conference on Multimedia Computing and Systems, pp. 201–208 (1996)
3. Shen, H.T., Ooi, B.C., Tan, K.L.: Giving meanings to WWW images. In: ACM Multimedia, pp. 39–48 (2000)
4. Srihari, R.K., Zhang, Z., Rao, A.: Intelligent indexing and semantic retrieval of multimodal documents. Information Retrieval 2, 245–275 (2000)
5. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(12), 1349–1380 (2000)
6. Haralick, R.M., Shanmugam, K., Dinstein, I.: Texture features for image classification. IEEE Transactions on Systems, Man, and Cybernetics 3(6), 610–621 (1973)
7. Tamura, H.: Textural features corresponding to visual perception. IEEE Transactions on Systems, Man, and Cybernetics 8(6), 460–473 (1978)
8. Kruskall, J., Liverman, M.: The symmetric time warping problem: from continuous to discrete. In: Time Warps, String Edits and Macro-molecules: The Theory and Practice of Sequence Comparison, pp. 125–161. Addison-Wesley, Reading (1983)
9. Edmonds, A.N.: Time Series Prediction Using Supervised Learning and Tools from Chaos Theory. PhD Thesis, University of Luton (1996)
10. Casdagli, M.: Nonlinear prediction of chaotic time series. Physica D Nonlinear Phenomena 35, 335–356 (1989)
11. Gu, S., Wang, Z., Chen, J.: The fractal research and predicating on the times series of sunspot relative number. Applied Mathematics and Mechanics 20(1), 84–89 (1999)
12. Jackson, E.A.: Perspectives of Nonlinear Dynamics. Cambridge University Press, Cambridge (1989)
13. Abarbanel, H.D.I., Brown, R., Sidorowich, J.J., Tsimring, L.S.: The analysis of observed chaotic data in physical systems. Reviews of Modern Physics 65, 1331–1392 (1993)
14. Fraser, A.M., Swinney, H.L.: Independent coordinates for strange attractors from mutual information. Phys. Rev. A 33(2), 1134–1140 (1986)
15. The Multiple-Dimensions Mutual Information Program (Matt Kennel), `http://www-ncsl.postech.ac.kr/en/softwares/archives/mmi.tar.Z`
16. Takens, F.: Detecting strange attractors in turbulence. In: Dynamical Systems and Turbulence, pp. 366–381. Springer, Heidelberg (1980)
17. Mañé, R.: On the dimension of the compact invariant sets of certain nonlinear maps. In: Dynamical Systems and Turbulence, pp. 230–242. Springer, Heidelberg (1980)
18. Medio, A., Gallo, G.: Chaotic Dynamics: Theory and Applications to Economics. Cambridge University Press, Cambridge (1993)
19. Kennel, M.B., Brown, R., Abarbanel, H.D.I.: Determining embedding dimension for phase-space reconstruction using a geometrical construction. Phys. Rev. A 45(6), 3403–3411 (1992)
20. Picard, R., Graczyk, C., Mann, S., Wachman, J., Picard, L., Campbell, L.: MIT media lab: Vision texture database, `http://vismod.media.mit.edu/vismod/imagery/VisionTexture/`