Curvelets: a low-level framework for computer vision

S. Dekel and A. Sherman
GE Healthcare

Abstract: Curvelets are a recent construction of a tight frame that provides a stable representation of L_2 functions with the property of excellent time-frequency-orientation localization. In this work, we employ Curvelets as a computational framework for low-level vision and present experimental results. We believe that the mathematical properties of Curvelets make them a more suitable framework for computer vision than their predecessors: local Fourier transforms, Wavelets transform or steerable filters.

1 Introduction

Curvelets [3] are a recent construction of a tight frame that provides a stable representation of L_2 functions with the property of excellent time-frequency-orientation localization. There are also the closely related constructions of Contourlets [25] and Shearlets [14]. In [10], the authors show similarities between Curvelets and new models of the human vision system developed by researchers working in Natural Scene Statistics. So, the obvious question is: can Curvelets be actually used as the low-level vision building blocks for high-level computer vision algorithms? Indeed, usually the first part of a computer vision paper (e.g. [17], [19], [24], [26]) deals with low-level vision aspects: Which local filtering process to use? How to capture directionality? How to incorporate scale invariance? How to quantify saliency?

In this work we explain how Curvelets may serve as a unifying computational framework for low-level vision and present experimental results. Let us illustrate this using the low-level vision model in Figure 1. The Curvelets described in Section 2 play the role of the Receptive field, providing local directional linear responses to the visual data. Each of the algorithms of Section 3 solves a particular problem of computer vision by applying a nonlinear computational process based on the Curvelet coefficients. In accordance with the model of Figure 1, all our algorithms process neighborhood information at various scales, i.e, the local interaction between a Curvelet coefficient and coefficients whose essential support is 'close' to this coefficient.

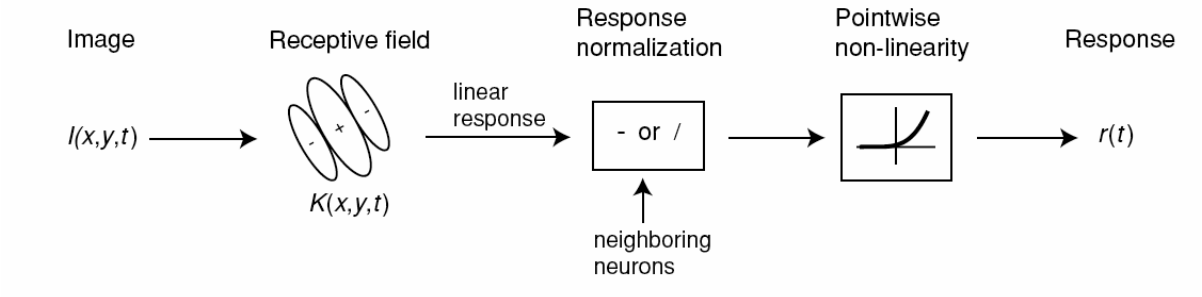


Figure 1. Model of the human low-level vision system (taken from [20])

2 Overview of Curvelets

2.1 Background on computational harmonic analysis tools in computer vision

There is a vast literature on the topic of harmonic analysis based systems for low-level vision and in this short section we can only review a few important highlights. In fact, many advances in modern harmonic analysis and in particular the constructions of localized time-frequency representations were inspired by vision research. Early examples are the Laplacian pyramid of Burt and Adelson [2] and the Marr-Hildreth Mexican hat edge detector [23]. Their modern descendents, Wavelets [11, 21], provide excellent time-frequency localization, perfect reconstruction and are a good platform for sparse representation of signals. However, Wavelet capture well 'point singularities', which is optimal in the univariate case, but only sub-optimal in higher dimensions, since further processing is required to extract local directional information from the coefficients. This is particularly true for the critically sampled Wavelet transforms. The complex wavelets [18] also provide perfect reconstruction, but improve upon the critically sampled Wavelet transforms by better capturing directionality and also support shift-invariance. The Wavelet maxima transform [22] is a Wavelet-based model, that is perhaps more suitable for computer vision algorithms due to its highly 'over-sampled' representation. It is related to edge detection by taking the local maxima of its responses.

A system that is able to capture directionality is the Steerable filters [12, 27] which have been applied extensively in computer vision. However, they do not provide perfect reconstruction and in applications one needs to determine their sampling rate, i.e., how many directions and locations at given scales. There are also other constructions of systems of low-level vision 'primitives' [29]. In comparison, Curvelets, by construction, have the 'right' amount of increase in the sampling rate with higher scales. This precise sampling rate is derived by tiling of the frequency domain into wedges and leads to unique approximation theoretical properties, such as efficient representation of edges, which in some sense is optimal.

2.2 Properties of Curvelets

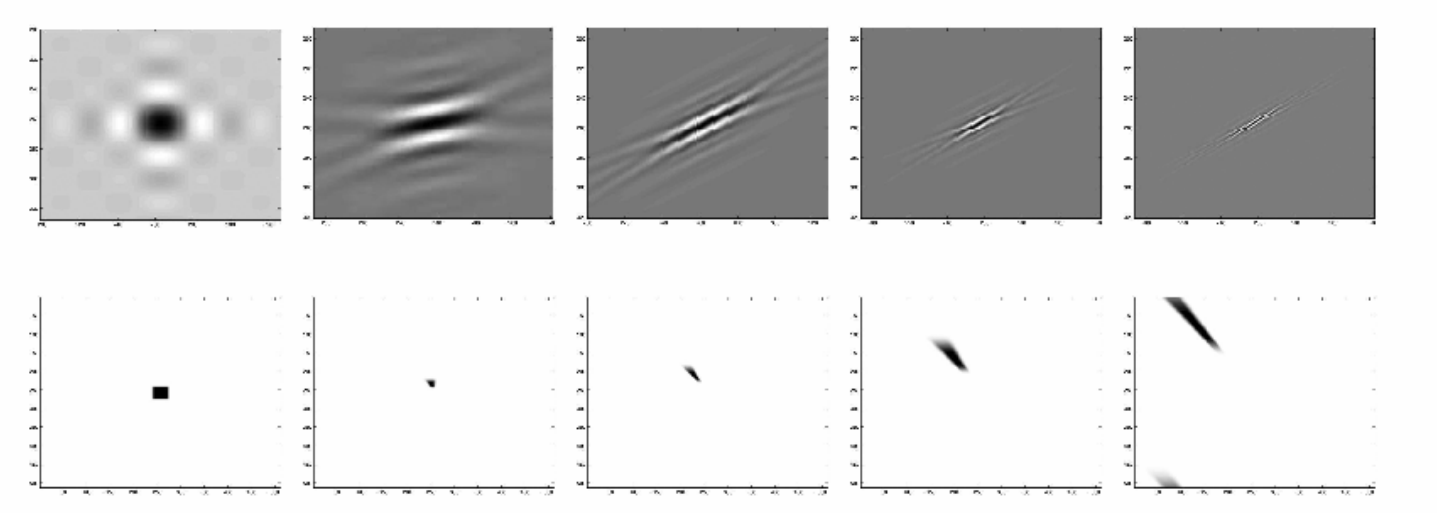
Curvelets are a tight frame of $L_2\left(\mathbb{R}^2\right)$. Each function $f \in L_2\left(\mathbb{R}^2\right)$ has the representation

$$f = \sum_{j,l,k} \left\langle f, \varphi_{j,l,k} \right\rangle \varphi_{j,l,k} ,$$

where $j \geq 0$ is the scale index, $l \in [0, 2\pi]$ is the orientation index and $k \in \mathbb{Z}^2$ is the location. Also, the Parseval equality holds

$$\|f\|_2^2 = \sum_{j,l,k} \left| \left\langle f, \varphi_{j,l,k} \right\rangle \right|^2 .$$

In Figure 2 we see some examples for Curvelets at various scales and directions. In Figure 2(a) we see that the Curvelet functions at the scale 0 are of different nature, exactly as in the Fourier and Wavelets transforms and their role is to capture a (very) low-resolution approximation of the function. From the scale $j = 1$ and going to higher scales, the essential support of the Curvelet functions becomes longer and thinner.



(a) $j = 0$ (b) Several scales, $j \geq 1$

Figure 2. The Curvelet frame. Top: Real-part of Curvelet functions at several scales. Bottom: the corresponding modulus of the Fourier transform of each Curvelet. (Taken from [6]).

Curvelets have a number of important features of that play a crucial role in applications requiring a sparse representation of the visual data:

1. As demonstrated in Figure 2, Curvelets are well-localized in frequency and orientation since they supported on specific wedges of the frequency domain.
2. Curvelets are also well-localized in the time domain. At the scale $j \geq 1$ they are essentially supported on ellipses of length $2^{-j/2}$ and width 2^{-j} and have rapid decay. This specific parabolic ratio implies that Curvelets are, in some approximation-theoretical sense, almost optimal (see property 5).
3. Observe that since Curvelets are constructed by tiling of the frequency plane, they are C^{∞} complex functions.
4. Curvelet have infinite number of moments and also 'directional' moments, i.e.

$$\int_{-\infty}^{\infty} \varphi_j(x_1, x_2)\, x_1^k dx_1 = 0, \qquad k \geq 0,\ \forall x_2\ , \tag{2.1}$$

holds with φ_j the 'generating' Curvelet for the scale $j \geq 1$, whereas all wavelets at this scale correspond to rotations and shifts of φ_j (see the details in [3]). These properties imply that if a Curvelet is essentially supported on a smooth part of the image, then the coefficient's modulus is relatively small. When a Curvelet is aligned with an edge of the image, then the modulus of its coefficient will be significant. In the case the Curvelet is essentially intersecting an edge, but not aligned with the edge, then the size of the coefficient depends on the local angle between the direction of the Curvelet function and the edge (see the detailed micro-local analysis of [5]).
5. Assume we wish to approximate $f(x) = \mathbf{1}_{\Omega}(x)$, the indicator function of a smooth compact domain $\Omega \subset [0,1]^2$. In this special case, one can approximate the boundary $\partial\Omega$ using a polygon with segments of

length $1/N$ and accuracy $O\left(1/N^2\right)$. Based on the boundary curve approximation, one can construct a triangulation $[0,1]^2 = \bigcup_{n=1}^{CN} \Delta_n$ (the exact procedure can be found in [7, Section 3]) and an approximation

$$S_N(x) = \sum_{n=1}^{CN} c_n \mathbf{1}_{\Delta_n}(x), \qquad c_n := \begin{cases} 1 & \Delta_n \cap \Omega \neq \varnothing, \\ 0 & else, \end{cases}$$

that satisfies

$$\|f - S_N\|_2 \leq C(\Omega)\frac{1}{N}. \tag{2.2}$$

Now let f_N be the nonlinear (greedy) approximant

$$f_N(x) := \sum_{n=1}^{N} \left\langle f, \varphi_{j,l,k}^{(n)} \right\rangle \varphi_{j,l,k}^{(n)}, \quad \text{where} \quad \left|\left\langle f, \varphi_{j,l,k}^{(1)} \right\rangle\right| \geq \left|\left\langle f, \varphi_{j,l,k}^{(2)} \right\rangle\right| \geq \cdots. \tag{2.3}$$

Then, it is shown in [4] that

$$\|f - f_N\|_2 \leq C(\Omega)\frac{\left(\log(N)\right)^{3/2}}{N}. \tag{2.4}$$

We note that the parabolic scaling of Curvelets' essential width and length comes into play in (2.4). Therefore, the performance of nonlinear Curvelet approximation (2.4) is near-optimal, since it is, up to a logarithmic factor, equivalent to the estimate (2.2) of the curve based method. In Figure 3(a), the function f is a digital x-ray of an hand and in Figure 3(b) and Figure 3(c) we see two examples for sparse approximations of the type (2.3) with $N = 4000$ and $N = 1500$, respectively.

(a) **(b)** **(c)**

Figure 3. Nonlinear approximation using Curvelets. (a) f - Original 512×512 CR image, (b) Approximation using f_{4000} (1:64 ratio), (c) Approximation using f_{1500} (1:175 ratio)

We conclude this section by noting the following a 'limitation' of Curvelets in their current form [3]. In similar way to the Wavelet transform, Curvelets are almost 'critically' sampled. This property is important for compression applications, but for some computer vision applications, such as object recognition (see Section

3.3) it is sometimes advantageous to allow a more redundant representation. For example, in order to identify more precisely locations of significant features, it might be necessary to have more orientations at a given scale or 'sample' a given scale and orientation at more centers. Control over scales and orientations can be easily done by changing the tiling of the frequency planes. However, adding more locations might imply that this over-redundant Curvelet system will become a frame instead of a tight frame and that a second dual Curvelet system needs to be employed for the purpose of computing function representations.

3 Computer vision applications

In this section we give details on applications of Curvelets in several fundamental computer vision tasks. In applications, the Fast Discrete Curvelet Transform (FDCT) is applied using a Cartesian grid of the time and frequency domains [3], to better suit the rectangular geometry of input images. In this work, we used the MatLab code for the FDCT published online at [6]. Thus, for each input digital image I we compute using the FDCT a representation as follows

$$\left\{I\left(x_1, x_2\right)\right\} \underset{FDCT}{\Rightarrow} I_{low} + \left\{c_{j,l,k} = \left\langle I, \varphi_{j,l,k} \right\rangle\right\}_{j \geq 1}. \tag{3.1}$$

The algorithm's complexity is $O\left(n^2 \log n\right)$ for an image of dimension $n \times n$. Here are some common properties of all of the computer vision methods described in this section:

1. In all of our methods we do not use the highest scale Curvelet coefficients, for several reasons:
    a. We avoid the stability issue resulting from sampling thin wedges in the frequency domain [Section 7.1, 3].
    b. In this work we do not target detection or understanding of very small features at high resolution, but rather 'visual analysis' of the main features and components of the image.
    c. This speeds up the computations.
2. The coefficients $\left\{c_{j,l,k}\right\}$ are, in general, complex numbers. In all our applications, we actually use their modulus, which correspond to local responses to visual activity at a certain scale, orientation and location of the image. This implies that in all our algorithms, at each scale, we only require the Curvelet coefficients corresponding to the first half of the orientations, since, by construction, the modulus of the Curvelets coefficients at the same location with orientation difference of π is identical.
3. At no point do we require the inverse Curvelet transform. Once the visual data is transformed to the Curvelet domain, all the analysis is done using the modulus of the coefficients. We note that the inverse transform is more computationally intensive than the forward transform.

Remark Note that we only use in our algorithms only partial Curvelet information. This gives a motivation to create a faster discrete Curvelet analysis scheme computing only the essential Curvelet data, i.e., only certain low scales, only half the orientations and only the modulus of coefficients.

We now review several computer vision applications. The common basis for all these method is that they are applied to the Curvelet representation of a given image.

3.1 Content-based image retrieval

Content-Based Image Retrieval (CBIR) is a well researched field. All CBIR algorithms are essentially tools that, given query image, are used to perform a fast search in an image database and output the most 'similar' images. A key observation is that each method uses a different notion for the 'similarity' between two images and the performance of the algorithm is measured in this context. It is standard to require that the search algorithm is invariant under shift, 2D rotation and scaling of the main 'object' in the image. Sometimes, there are more demanding requirements such as invariance under color and illumination changes and even 3D rotation of the given object.

Here, we consider a specific medical imaging application. We assume that there exists a huge database containing images acquired via Computed Tomography (CT), Magnetic Resonance Imaging (MRI), Ultrasound (US), Positron Emission Tomography (PET), Computed Radiography (CR), etc. We consider the scenario where a physician diagnosing a patient and observing a new result of an imaging exam wishes to fetch from the database visually similar images that have been already pre-diagnosed by himself/herself or others. In most cases, the acquisition method and the body part are known and written into the DICOM (standard format for medical images) header. In Figure 4, we see an example for a query image. The DICOM header of this image contains the field-tag values: Modality = CR, Body part = NECK.

Figure 4. Input neck spine x-ray image

Using these tags, one can perform textual-based pre-filtering of the database and extract only the subset of images with the same tag-field values. In Figure 5 we see samples from the image database, all of which have the same tag-field values except for one head x-ray, whose body-part field-tag was left empty by the lab-technician at the time of the acquisition.

Figure 5. Samples from medical image database

In Figure 6 we see the two top-matches (from the samples in Figure 5) computed by our Curvelet-based CBIR algorithm, equipped with a reflection-invariance feature.

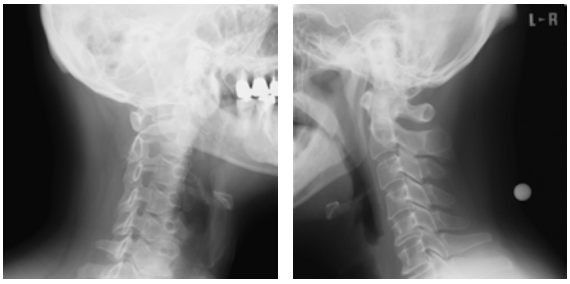


Figure 6. Top matches using our Curvelet-based CBIR algorithm (with reflection invariance).

We now provide some details of our simple algorithm. We follow a classic retrieval paradigm: First we compute from the image a ***signature***, i.e., a vector of real numbers whose length is overwhelmingly smaller than the number of pixels in the image. The vector should be, in principal, invariant to shift, rotation and possibly scaling of the main visual features in the image. Also, for two images which are 'similar' but one is a deformation of the other, we expect the signature to be also similar. We then compare the signature with pre-computed signatures of images in the database to find the top matches. Choices for signatures that are frequently found in the literature are various types of histograms of pixel values or responses of filters designed to detect edges and textures in the image [26]. These methods capture in the signature the 'amount' or 'nature' of edges or texture, but do not capture the structure. As one can understand from Figure 5, this approach is not applicable in our medical imaging setup, since almost all images of a specific body-part, acquired using the same imaging procedure (in this case CR), have 'similar' overall features and differ only in the geometry of the structure. Nevertheless, one can consider, in some cases, to use histograms generated from the low-resolution images I_{low}, produced by the Curvelet transform (3.1), for fast 'rough' pre-filtering. We also note that some Wavelet-based methods [16, 32] do not satisfy the requirements of shift, rotation and scale invariance.

To better capture structure invariant under shift, rotation and scale, our method utilizes Hu's invariants [15], a method that is known to work well in the simpler framework of segmentation-based CBIR methods and also for Wavelet Maxima-based CBIR [8]. Assume for simplicity that $g:[0,1]^2 \to \mathbb{R}$ is very 'sparse' in the sense that in attains large values only on a very small subset and is also compactly supported in some domain $\Omega \subset [0,1]^2$. We compute normalized moment

$$\overline{x}_1 := \frac{\int_{\mathbb{R}^2} x_1 g(x_1,x_2)\,dx}{\int_{\mathbb{R}^2} g(x_1,x_2)\,dx}, \qquad \overline{x}_2 := \frac{\int_{\mathbb{R}^2} x_2 g(x_1,x_2)\,dx}{\int_{\mathbb{R}^2} g(x_1,x_2)\,dx}$$

$$\mu_{p,q} := \int_{\mathbb{R}^2} (x_1-\overline{x}_1)^p (x_2-\overline{x}_2)^p\, g(x_1,x_2)\,dx\,. \tag{3.2}$$

Hu's invariants remain unchanged under shift, rotation, scale of $g(x)$. Here are the definitions of the first three

$$\mu_{2,0}+\mu_{0,2}\,,$$
$$(\mu_{2,0}-\mu_{0,2})^2+4\mu_{1,1}^2\,,$$
$$(\mu_{3,0}-\mu_{1,2})^2+(\mu_{2,1}-\mu_{0,3})^2\,.$$

To apply Hu's invariants in the Curvelet framework we compute multiresolution edge maps from the Curvelet transform as follows. We fix a lowest scale $j_1 \geq 1$ and highest scale $1 \leq j_1 \leq j_2$ and for each $j_1 \leq j \leq j_2$ we set

$$g_j(x_1,x_2) := \frac{\tilde{g}_j(x_1,x_2)}{\max \tilde{g}_j}, \qquad \tilde{g}_j(x_1,x_2) := \max_{v_{j,l,k}=(x_1,x_2)} \left|c_{j,l,k}\right|, \quad v_{j,l,k} \text{ center of } \varphi_{j,l,k}, \tag{3.3}$$

where it is understood that $\tilde{g}_j(x_1,x_2) = 0$ at points (x_1,x_2) that are not centers of any Curvelets. Note that $g_j(x_1,x_2)$ is indeed a well defined function. For example, if (x_1,x_2) is a corner point where several Curvelets have their centers, the maximum in (3.3) will be attained at one of the orientations aligned with a smooth curve segment ending at the corner. The normalization in (3.3) is needed, since we want to compare different scales and images with different brightness levels. We note that our algorithm heavily relies on the approximation properties of Curvelets and our heuristics are that only a small fraction of the Curvelet coefficients are significant and that theses significant coefficients capture local features. Therefore, the model we have in mind is that $g_j(x_1,x_2)$, $j_1 \leq j \leq j_2$, are sparse maps, similar in nature to segmentation maps. In Figure 7 we see a visualization of the Curvelet-based edge map $g_5(x_1,x_2)$ associated with the Curvelet coefficients at the scale 5.

Figure 7. Lena image and the visualization of the Curvelet edge map $g_5(x_1,x_2)$

Next, we compute the j th-scale signature $v_j^q \in \mathbb{R}^7$ as Hu's invariants for the normalized Curvelet response function $g_j(x_1,x_2)$. The signature of a given image is the union of all scale signatures $Sig_I := \{v_j^q\}_{j=j_1}^{j_2}$. For a given distance ρ on $\mathbb{R}^7$, we define the correlation between a query image I_q and a database image I_{db} as

$$\text{auto}(Sig_q, Sig_{db}) := \min_{\substack{j_1 \leq j_q < j_2 \\ j_1 \leq j_{db} < j_2 \\ 0 \leq w \leq 1}} \left\{ w\rho\left(v_{j_q}^q, v_{j_{db}}^{db}\right) + (1-w)\rho\left(v_{j_q+1}^q, v_{j_{db}+1}^{db}\right) \right\} \tag{3.4}$$

We emphasize that we have selected a very simple approach to fast CBIR using relatively small 'signatures'. We believe that a promising line of research is to apply more advanced CBIR methodologies incorporating learning [29], using Curvelets as the underlying low-level system.

3.2 Texture analysis and retrieval

In this section we discuss the application of Curvelets to texture analysis and retrieval. Given a query texture patch we wish to find a corresponding match from a texture database. In Figure 8 we see texture samples from the Brodatz database [1] (the digital samples are taken from [31]).

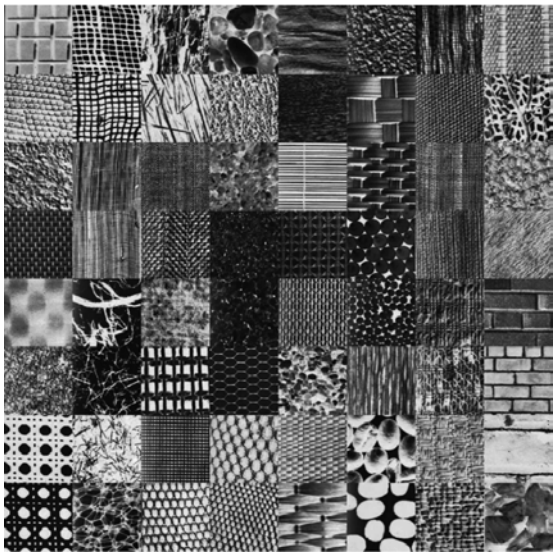


Figure 8. Images from the Brodatz database

As in the CBIR application, we would like our method to be invariant under shift, rotation and scale. There is extensive literature on transform based texture analysis, however most of the published methods do not support either rotation or scale invariance (or both). Essentially, each method transforms the image using Local Fourier, Gabor, Wavelets or steerable filters and then applies statistical models to the response of the wave functions. Many of the statistical models applied in the Wavelet framework (e.g. [9]) are fairly complex. This is possibly due to the fact that Wavelets do not capture directionality too well and so one needs to search for elaborate statistical patterns and relationships between neighbor coefficients at the same scale and/or 'ancestor' coefficients, i.e. coefficients at lower scales whose support contains the support of the corresponding higher resolution coefficient. In [25] the authors describe texture analysis using Contourlets which are very similar in nature to Curvelets. However, the statistical processing in [25] applied to the transform coefficients is too reminiscent of 'direction-less' wavelet statistics. Perhaps the closest approach to what we present below are the rotation invariant methods of [13] and [30], where steerable filters are used as the underlying low-level system.

In this work we model texture using matrices $M_j(l_1,l_2)$, $1 \le j_1 \le j \le j_2$, where each matrix entry represents local correlation between the two directions indexed by l_1 and l_2 at the given scale j. Thus, textural information is captured by a relatively small multiscale texture 'signature' composed of a collection of small matrices. In the discrete implementation of the Curvelet transform, each scale j contains Curvelets in the (Cartesian analogue of) angles

$$\theta_l = 2\pi 2^{-\lfloor j/2 \rfloor} l, \qquad l = 0,1,\ldots, \qquad 0 \le \theta_l < \pi.$$

Let $v_{j,l,k}$ be the 'center' of the Curvelet function $\varphi_{j,l,k}$ and let ρ be some distance on $\mathbb{R}^2$ (e.g. the l_p distance for $1 \le p < \infty$, where l_2 is the Euclidian distance). We quantify the local 'interaction' between two directions, at the scale j using the following formula

$$M_j(l_1,l_2) := \frac{\tilde{M}_j(l_1,l_2) - \min \tilde{M}_j}{\max \tilde{M}_j - \min \tilde{M}_j}, \qquad \tilde{M}_j(l_1,l_2) := \sum_{k_1} p_{j,l_1,k_1} \sum_{k_2} d_{j,l_1,k_1}(l_2,k_2)\, p_{j,l_2,k_2},$$

with (3.5)

$$p_{j,l,k} := \frac{|c_{j,l,k}|}{\sum_{k'} |c_{j,l,k'}|}, \qquad d_{j,l_1,k_1}(l_2,k_2) := \frac{\overbrace{\left(\rho\left(v_{j,l_1,k_1}, v_{j,l_2,k_2}\right)+1\right)^{-1}}^{\text{how 'far' is } \varphi_{j,l_2,k_2} \text{ from } \varphi_{j,l_1,k_1}}}{\underbrace{\sum_{k}\left(\rho\left(v_{j,l_1,k_1}, v_{j,l_2,k}\right)+1\right)^{-1}}_{\text{normalization term}}}.$$

Observe that $M_j(l_1,l_2)$ is simply a normalization of $\tilde{M}_j(l_1,l_2)$. Each entry of $\tilde{M}_j(l_1,l_2)$ receives major contributions in cases where there are two significant Curvelet coefficients at the scale j and directions l_1 and l_2 with 'close' centers. In cases where one of the coefficients is not significant or the two coefficients correspond to Curvelets whose centers are relatively far, the contribution is small. Let us demonstrate this principal with a simple example. In Figure 9(a) we see an image of a black square and in Figure 1(b) and Figure 1(c) we see color maps of the entries of $M_2(l_1,l_2)$ and $M_3(l_1,l_2)$, respectively.

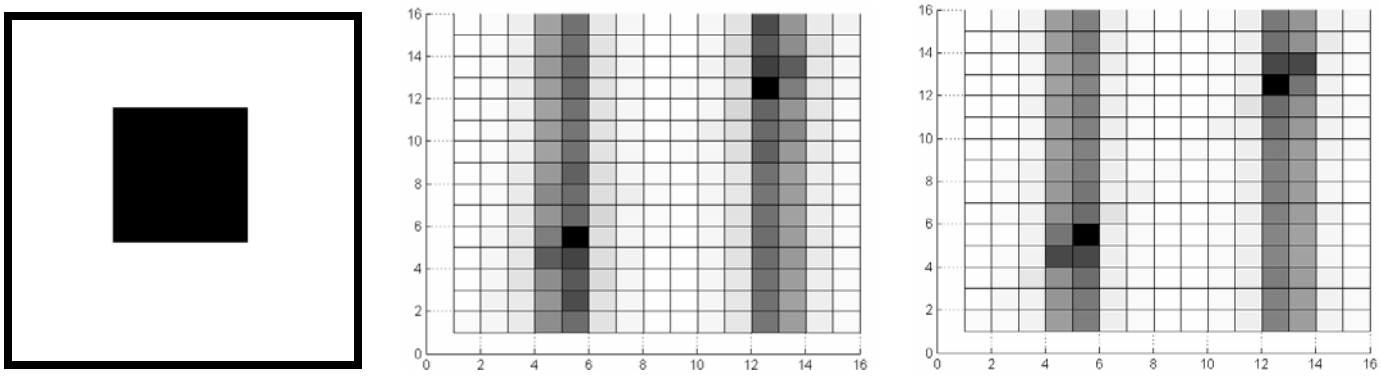
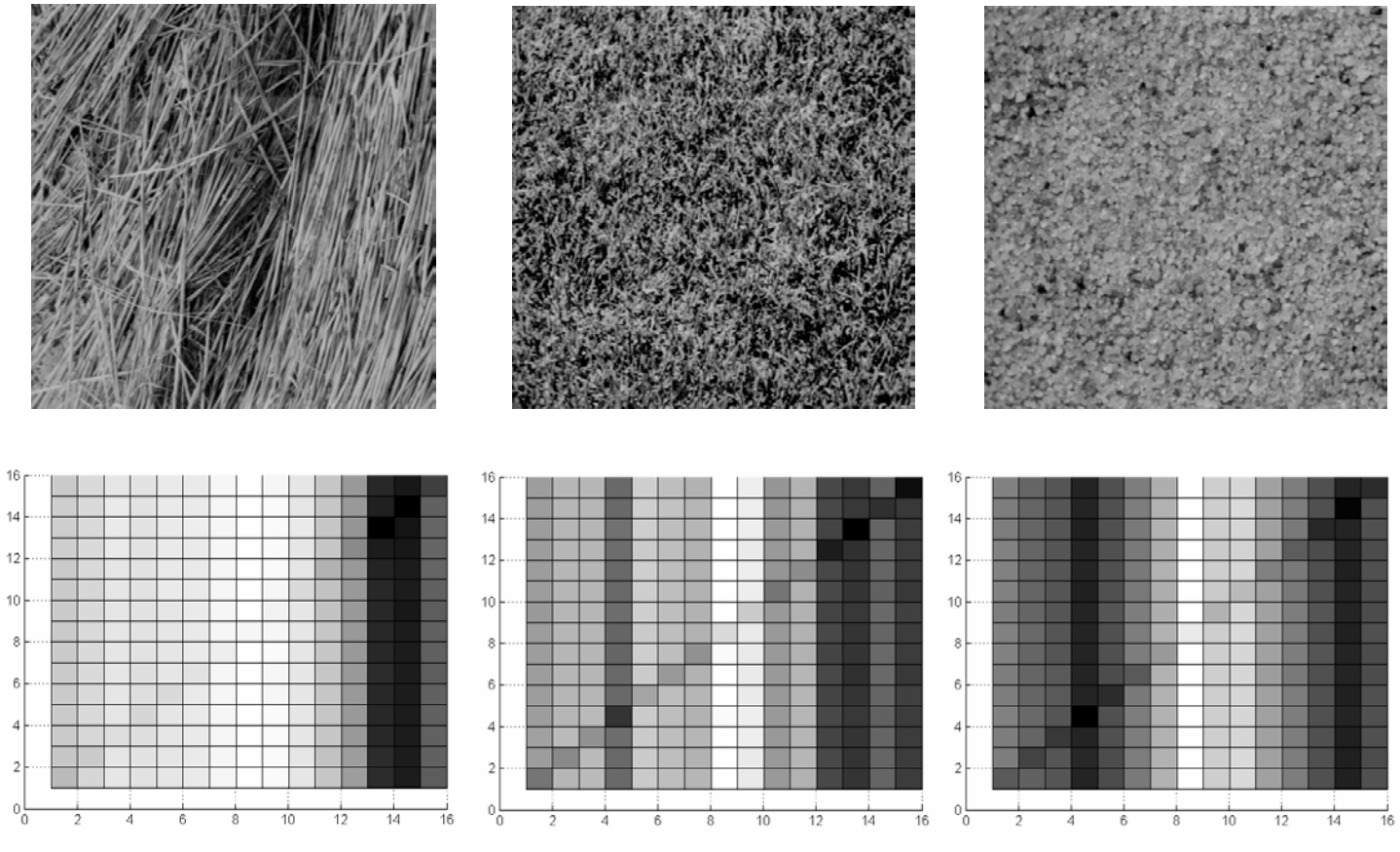


Figure 9 (a) Image of a black square, (b) The matrix $M_2(l_1,l_2)$, (c) The matrix $M_3(l_1,l_2)$

At both scales, we see two significant entries on the diagonal at the directional indices corresponding to the horizontal and vertical directions which are indeed the orientations of the edges of the square. However we see some local interaction between these two main directions and all other directions. This comes from the Curvelet coefficients whose essential support is concentrated in the vicinity of the black square's corners (see [5] for detailed analysis of the behavior of the continuous Curvelet transform at point, curve and corner singularities). Notice that due to normalization, $M_2(l_1,l_2)$ and $M_3(l_1,l_2)$ are very similar, but $M_3(l_1,l_2)$, the matrix

corresponding to the higher scale better captures the information, i.e., it has 'higher peaks' at the two main orientations and less interaction between orientations at corners.

In Figure 10 we see examples of directional interaction matrices for several texture samples. The reader should interpret the matrix $M_3(l_1, l_2)$ for the Straw image (Figure 10(a)) as follows: the texture has strong directionality in a specific orientation, but with strong local interactions with all other directions. Notice how our simple local analysis reveals the oriented structure in the Sand image (Figure 10(c)) which corresponds to the shape of the grains.



(a) Straw **(b) Grass** **(c) Sand**

Figure 10. Texture test images and the corresponding matrices $M_3(l_1, l_2)$ computed by (3.5)

In Figure 11 we demonstrate that our local directional analysis reveals the very different structure for textures that visually seem very similar.

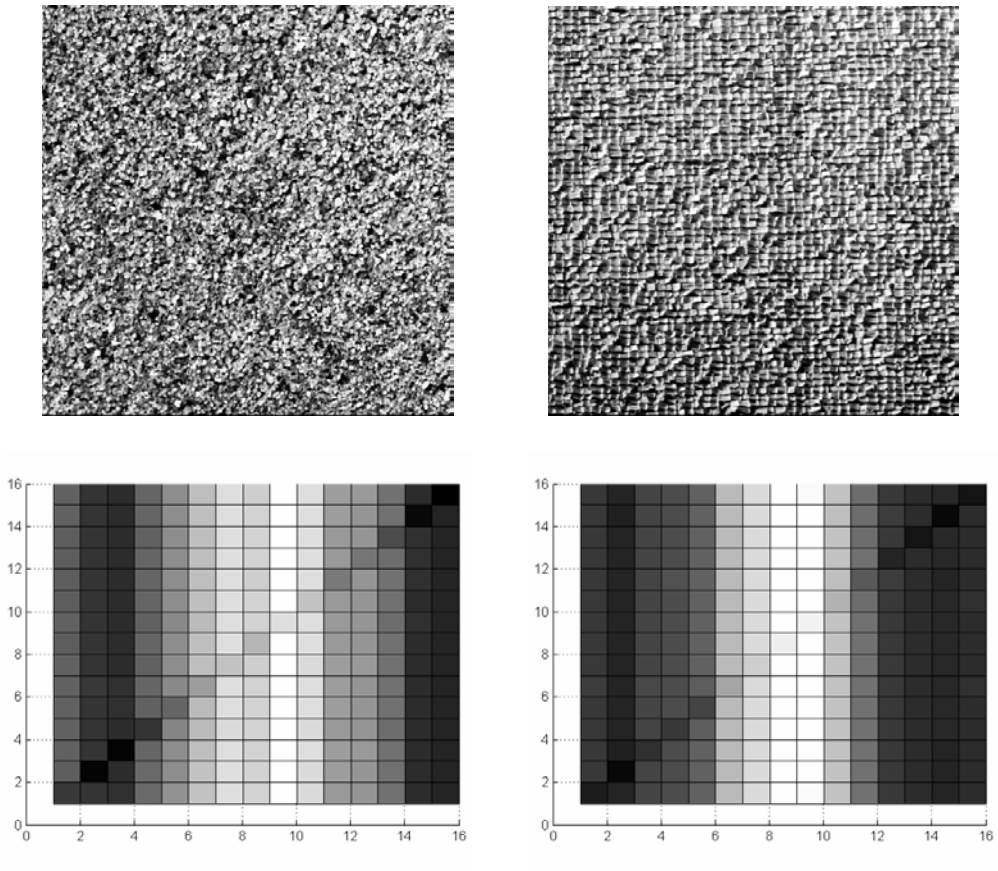
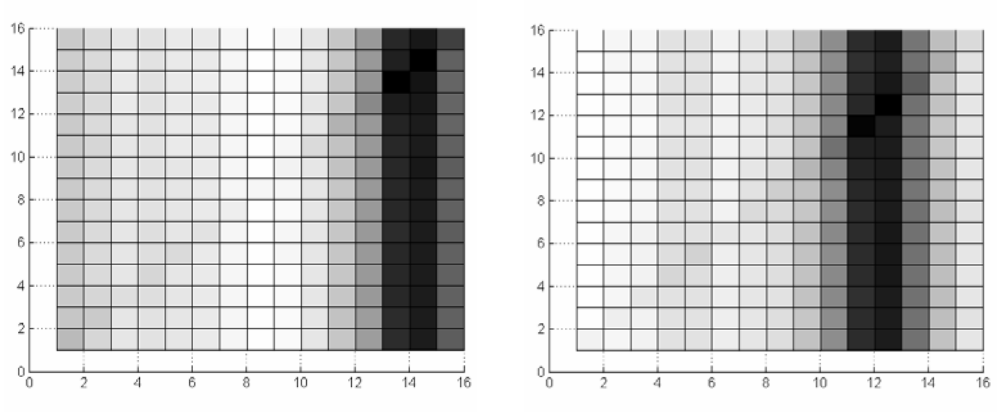
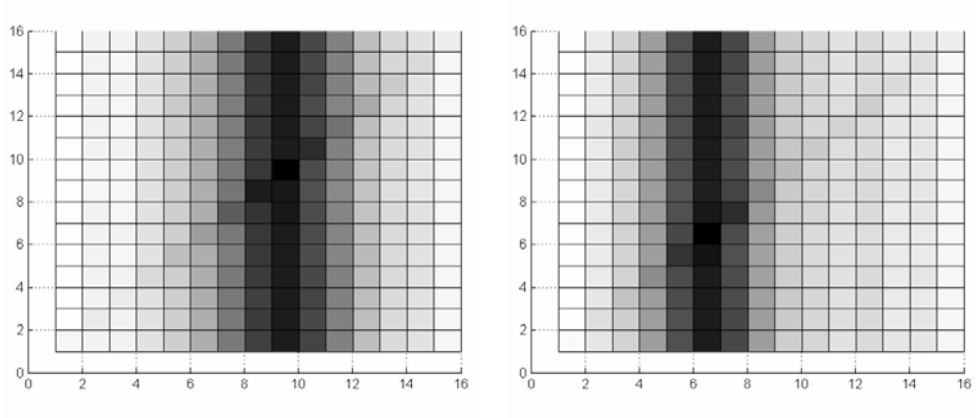


Figure 11. Visually similar textures and their very different local directionality matrices

In Figure 12 we see the matrices $M_3(l_1, l_2)$ for rotated versions of the straw image by 0,30,60 and 90 degrees. Observe that for each rotation the directional correlation matrix is approximately a shifted version of the original Straw image in the direction of its second diagonal (top-left to bottom-right).



(a) 0 degrees (b) 30 degrees



(c) 60 degrees (d) 90 degrees

Figure 12. The matrices $M_3(l_1,l_2)$ for rotations of the straw image are roughly periodic along the second diagonal.

the matrix M along the second diagonal corresponding to a rotation $0 \le \theta < \pi$. Then, one way to quantify the distance between two texture images f_1 and f_2, by employing their corresponding directional texture matrices $\{M_j^1\}$ and $\{M_j^2\}$, is to calculate

$$\min_{\substack{0\le\theta<\pi \\ j_{\min}\le j_1, j_2 < j_{\max}}} \left(\tilde{\rho}\left(M_{j_1}^1, M_{j_2}^2(\theta)\right) + \tilde{\rho}\left(M_{j_1+1}^1, M_{j_2+2}^2(\theta)\right)\right), \tag{3.6}$$

Let us see how invariance is achieved by the minimization (3.6). The individual matrices $\{M_j\}$ are already shift invariant. Rotation invariance is achieved by minimizing over the angle θ, while scale invariance is supported by minimizing over scale correspondences $j_1 \leftrightarrow j_2$. For scale invariance it is important that the matrices $\{M_j\}$ are normalized. As can be seen from the examples we provide, for 'rough' analysis even one low resolution is enough, so long that the coefficients at this scale are processed correctly (i.e. in similar manner to (3.5)).

3.3 Object recognition

Given an image of a certain object, the goal of an Object Recognition (OR) method is to identify if and where the given object appears in any given input image. Reasonable OR algorithms should be at least invariant under shifts, rotations, change in size (scale) and possibly some partial occlusion of the given object in the input images. That is, if the object does appear at any location in the tested image, possibly rotated, possibly smaller or bigger and possibly partially occluded by other objects, then the OR method should return a positive answer with an identification of the location(s) of the object.

In our Curvelet-based OR method we follow the methodology of local feature matching. This approach has been used before although differently (e.g. [24]). The main idea is that the object contains several key features and that, if the object exists in the input image then there exists a single affine transform that maps them to features in the input image. We define a local feature as a local group of significant Curvelet coefficients at a given scale. This approach allows great flexibility where the characteristics of local features may depend on the type of objects that we need to recognize and on performance requirements. Observe that the number of local features in the input image is, in general, significantly larger as the number of object features,

since the object features (if the object exists in the input image) are only a subset of the total features in the input image.

We choose to identify local features with 'local' groups of Curvelets satisfying the following conditions:

(a) The number of Curvelets in the group is bounded from below and above using pre-determined thresholds. The idea is that a local feature will describe a reasonably significant curved edge piece. However, for the purpose of supporting occlusions, the pieces should not be too long. In our experiments below we used a Curvelet groups of size 3.
(b) The Curvelet coefficients' moduli are above some threshold.
(c) The Curvelets' centers are sufficiently close (relative to their scale).
(d) The Curvelets' orientation is similar (relatively their scale). Observe that this can be done by determining neighborhoods of wedges.

To achieve the goal of scale invariance, we pre-compute local 'edge piece' features for the object image at a few scales. To improve the performance of our algorithm, for each input image we compute only the local features corresponding to a certain fixed scale (we used the scale 2 in our experiments).

For each scale of the object image we proceed as follows. Recall that an affine transform A on $\mathbb{R}^2$ is determined by a 2x2 matrix M and shift vector $v \in \mathbb{R}^2$, such that $Ax = Mx + v$, for each point $x \in \mathbb{R}^2$. For each three object local features and three input local features, we compute the unique affine transform that maps the object feature centers to the input feature centers. We allocate a grade/weight to the match of the two threesomes by the correspondence in directionality between the features.

The set of all computed affine transforms determines a bounded domain in $\mathbb{R}^6$. We quantize this bounded domain using a cover with overlapping centers. Each 'sample' affine transform in our quantization receives a score based on the sum of weights of the correspondence affine transforms in its neighborhood. Once the affine transform with the highest score is found, we check to see if the score is bigger than some pre-defined threshold. The threshold corresponds to the minimal number of features we want to match and the tolerance for the approximation of the match. If score is bigger than the pre-defined threshold, then we declare the object found and we can display the estimated location by marking the matched local features in the test image. In Figure 13 we see an object image of a road sign and in Figure 14 we see the output of our OR algorithm, correctly identifying the sign in the test image. Note that although the shapes of the sign object and sign in the test image are similar there is a significant contrast difference between the two.

Figure 13. Object image of a road sign

Figure 14. Test image with sign and Curvelet-based local feature matching produced by our algorithm.

References

1. P. Brodatz, Textures: A Photographic Album for Artists and Designers, Dover Publications, New York, 1966.
2. P. Burt and E. Adelson , The Laplacian pyramid as a compact image code, IEEE Trans. Communication 31 (1983), 532–540.
3. E. Candès, L. Demanet, D. Donoho, and L. Ying, Fast Discrete Curvelet Transforms, Multiscale Modeling & Simulation 5 (2006), 861-899.
4. E. Candès and D. Donoho , New tight frames of Curvelets and optimal representations of objects with piecewise C^2 singularities, Comm. Pure App. Math. 57 (2003), 219-266.
5. E. Candès and D. Donoho, Continuous Curvelet Transform: I. Resolution of the Wavefront Set, Appl. Comput. Harmon. Anal. 19 (2005), 162-197.
6. Curvelets web-site (http://www.curvelets.org).
7. S. Dekel, D. Leviatan and M. Sharir, On bivariate smoothness spaces associated with nonlinear approximation, *Constr. Approx*. 20 (2004), 625-646.
8. M. Do, S. Ayer and M. Vetterli, Invariant image retrieval using the wavelet maxima moment, Proceedings of 3rd Int. Conf. on visual info. and info. systems, 1999.
9. M. Do and M. Vetterli, Rotation invariant texture characterization and retrieval using steerable wavelet-domain hidden Markov models, IEEE Trans. Multimedia 4 (2002), 517-527.
10. D.L. Donoho and A.G. Flesia, Can recent innovations in harmonic analysis 'explain' key findings in natural image statistics?, Network: Computation in Neural Systems 12 (2001), 371-393.
11. D. Field, Wavelets, vision and the statistics of natural scenes, Phil. Trans. R. Soc. Lond. A 357 (1999), 2527-2542.
12. W. Freeman and E. Adelson, The design and use of steerable filters, IEEE Trans. Patt. Anal. and Machine Intell. 13 (1991), 891-906.
13. H. Greenspan, S. Belongie, R. Goodman and P. Perona, Rotation invariant texture recognition using a steerable pyramid, ICPR 1994, Jerusalem, Israel, 162-167.

14. K. Guo, G. Kutyniok, and D. Labate, Sparse Multidimensional Representations using Anisotropic Dilation and Shear Operators, Wavelets and Splines (Athens, GA, 2005), Nashboro Press, Nashville, TN (2006), 189-201.
15. M. Hu, Visual pattern recognition by moment invariance, IRE Trans. Info. Theory 8 (1962), 179-187.
16. C. Jacobs, A. Finkelstein and D. Salesin, Fast multiresolution image querying, SIGGRAPH Proc. 1995, 277-286.
17. T. Kadir and M. Brady, Saliency, Scale and Image Description, Comp. Vision 45 (2001), 83-105.
18. N. Kingsbury, Complex wavelets for shift invariant analysis and filtering of signals, Appl. Comput. Harmon. Anal. 10 (2001), 234-253.
19. D. G. Lowe, Object Recognition from Local Scale-Invariant Features, Proc. IEEE Comp. Vision 2 (1999), 1150-1157.
20. B. Olshausen and D. Field, How close are we to understanding V1?, Neural Comput. 17 (2005), 1665-99.
21. S. Mallat, Wavelets for a vision, Proc. IEEE 84 (1998), 604-614.
22. S. Mallat and S. Zhong, Characterization of signals from multiscale edges, IEEE Trans. Pattern Anal. Machine Intelligence 14 (1992), 710-732.
23. D. Marr and E. Hildreth, Theory of edge detection, Proc. R. Soc. Lond. B, 207 (1980), 187–217.
24. K. Mikolajczyk, A. Zisserman and C. Schmid, Shape recognition with edge-based features, Proceedings of the British Machine Vision Conference (2003).
25. D. Po and M. Do, Directional multiscale modeling of images using the contourlet transform, IEEE Trans. Image Processing 15 (2006), 1610-1620.
26. B. Schiele and L. Crowley, Recognition without correspondence using multidimensional receptive field histograms, Comp. Vision 36 (2000), 31-50.
27. E. Simoncelli and W. Freeman, The steerable pyramid: A flexible architecture for multi-scale derivative computation, in Proc. IEEE ICIP, Washington, DC, 1995.
28. E. Stein, Harmonic Analysis: Real-variable methods, Orthogonality and oscillatory integrals, Princeton University Press, 1993.
29. K. Tieu and P. Viola, Boosting image retrieval, Comp. Vision 56 (2004), 17–36.
30. G. Tzagkarakis, B. Beferull-Lozano and P. Tsakalides, Rotation-invariant texture retrieval with gaussianized steerable pyramids, IEEE Trans. Image Processing 15 (2006), 2702-2718.
31. USC-SIPI Image database (http://sipi.usc.edu/database/).
32. Z. Zhuang and M. Ouhyoung, Novel multiresolution metrics for content-based image retrieval, Proc. Fifth Pacific Conf. Computer Graphics and Applications 1997, 105-114.

Shai Dekel
GE Healthcare
Herzelia, Israel
Shai.dekel@ge.com
http://shaidekel.tripod.com

Alexander Sherman
GE Healthcare
Herzelia, Israel
Alexander.Sherman@ge.com