An Algorithm for Designing Multiple Gabor Filters for Segmenting Multi-Textured Images

Thomas P. Weldon
Dept. of Electrical and Computer Eng.
University of North Carolina at Charlotte
Charlotte, NC, USA 28223
E-mail: tpweldon@uncc.edu

William E. Higgins
Department of Electrical Engineering
Penn State University
University Park, PA, USA 16802
E-mail: weh2@psu.edu

Abstract

We present an algorithm for the design of multiple Gabor filters for the segmentation of multi-textured images. We draw upon earlier results that provide a segmentation error measure based on the predicted vector output statistics of multiple filter channels. This segmentation error measure is used to design the filter channels for a particular segmentation task. In our approach, the filter parameters are free to vary from channel to channel and are not restricted to some predetermined decomposition of the frequency plane. Thus, our method can generate more effective filter designs and result in more effective features for image segmentation than prior methods. Finally, we present texture segmentation results that confirm the efficacy of the proposed procedure. These results show effective segmentation of 8 textures using as few as 2 filters, whereas earlier approaches required 13 to 40 filters to segment 5 textures.

1. Introduction

We present a comprehensive procedure for the design of multiple filters for texture segmentation. Our approach permits greater flexibility in filter design and overcomes limitations in earlier methods. Although many investigators have successfully applied Gabor filters to the segmentation of multi-textured images [1–10], a comprehensive procedure for the design of multiple Gabor filters for texture segmentation remains largely an open issue. Drawing upon earlier results, we propose a new algorithm for the design of multiple Gabor filters to segment multiple textures. This algorithm overcomes limitations in prior *filter-bank* approaches, where the filters were selected from a limited set of predetermined candidate filters, and overcomes limitations in prior *filter-design* approaches, where the Gabor filters were designed for a particular texture-segmentation task.

Several investigators employed filter-bank approaches using wavelet and other similar filter-bank decompositions [2, 11–13]. These filter-bank approaches have sparse sets of candidate filters that provide fixed coverage of the frequency plane or fixed relationships between filter bandwidths and center frequencies. Other investigators have considered filter-design approaches. Bovik *et al.* [1] focused on the response of a single filter to a single texture rather than the general multi-filter multi-texture problem. Earlier work by the present authors [3, 4, 6] also focused on the design of single filters. These earlier filter-bank and filter-design approaches did not consider combined effects of Gabor filtering and subsequent Gaussian lowpass filtering, did not generate predicted vector output statistics of multiple filter channels, and did not provide a comprehensive mathematical framework for filter design. In addition, prior design methods used least-square reconstruction error or channel energy criteria for filter design rather than using predicted segmentation error as the design criteria [14].

To address these issues, we propose a new algorithm for the design of multiple Gabor filters to segment multi-textured images. This new algorithm overcomes the limitations found in earlier approaches and draws upon our recent results that provide the predicted multivariate output statistics of multiple filter channels and provide predicted segmentation-error as a filter-design basis [7, 8]. In our method, we first form an extensive set of candidate filters that effectively provide overlapping coverage of the frequency plane at multiple resolutions. We then use a forward-sequential procedure to iteratively design the multi-filter system, adding one filter at a time [14]. At each iteration, a vector measure of segmentation error is used to select the best filter when combined with the filters from prior iterations. Although the forward-sequential filter design procedure is sub-optimal, we find that our methods produce effective filter designs.

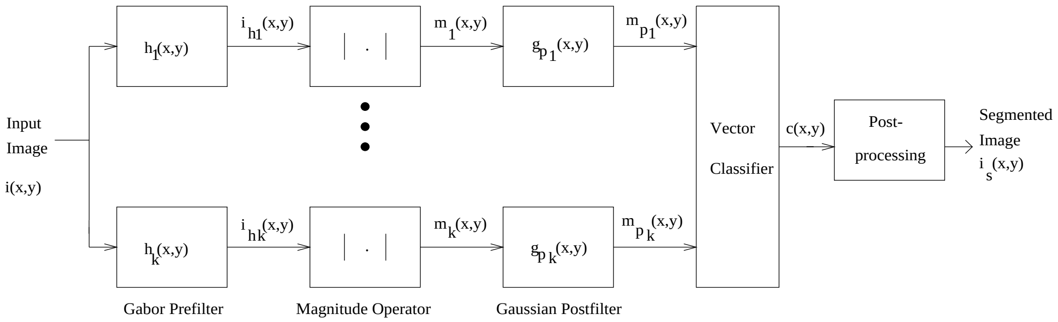

Figure 1: Multichannel system for segmenting a textured image.

In Section 2 we first review the image processing system used in our procedure and prior results on predicted segmentation error. We then describe the filter design algorithm in Section 3. Finally, we present results in Section 4 that demonstrate effective texture segmentation using the designed set of filters.

2. Multichannel scheme

Fig. 1 illustrates the image processing system considered in our design procedure. Further details can be found in [5–8].

In Fig. 1, the input image $i(x,y)$ is comprised of disjoint regions of $\mathcal{N}$ textures $t_1, t_2, \ldots, t_{\mathcal{N}}$ with $\mathcal{N} \geq 2$. This input is applied to k filter channels, where each channel is comprised of a bandpass *Gabor prefilter* $h_j(x,y)$, a magnitude operator $|\cdot|$, and a *Gaussian postfilter* $g_{p_j}(x,y)$. Typically, the number of channels k is less than the number of textures $\mathcal{N}$. The *Gabor prefilter* in channel j has impulse response $h_j(x,y)$:

$$h_j(x,y) \quad = \quad \frac{1}{2\pi\sigma_{g_j}^2}\, e^{-\frac{(x^2+y^2)}{2\sigma_{g_j}^2}} \quad e^{-j2\pi(u_jx+v_jy)} \quad (1)$$

where (u_j, v_j) is the center frequency of the filter, and a symmetric filter response is used for simplicity. The Gabor prefilter $h_j(x,y)$ has a frequency response

$$H_j(u,v) = e^{-2\pi^2\sigma_{g_j}^2[(u-u_j)^2+(v-v_j)^2]} \qquad (2)$$

where (u,v) is spatial frequency.

The output of the prefilter $i_{h_j}(x,y)$ is the convolution of the input image with the Gabor prefilter $i_{h_j}(x,y) = h_j(x,y) * i(x,y)$, where $*$ denotes convolution in two dimensions. The magnitude of the Gabor-prefilter output is then taken, $m_j(x,y) = |i_{h_j}(x,y)|$, where $m_j(x,y)$ has been shown to have approximately Rician statistics for filtered textures [5, 6, 8, 10].

A lowpass *Gaussian postfilter* $g_{p_j}(x,y)$ is then applied to $m_j(x,y)$ yielding the postfiltered image in the j^{th} filter channel $m_{p_j}(x,y) = m_j(x,y) * g_{p_j}(x,y)$ with

$$g_{p_j}(x,y) \quad = \quad \frac{1}{2\pi\sigma_{p_j}^2}\, e^{-\frac{(x^2+y^2)}{2\sigma_{p_j}^2}} \;, \qquad (3)$$

and where σ_{p_j} determines the Gaussian postfilter in the j^{th} channel. Thus, the parameters $\theta_j = (u_j, v_j, \sigma_{g_j}, \sigma_{p_j})$ completely determine filter channel j, and these parameters are free to vary from channel to channel. Matrix $\boldsymbol{\Theta_k} = [\theta_1, \theta_2, \ldots, \theta_k]^T$ determines the filter parameters of the k channels, where superscript T indicates transpose, and $\boldsymbol{\Theta_k}$ is a k row by 4 column matrix. We refer to $i_{h_j}(x,y)$ as the *prefiltered image*, $m_j(x,y)$ as the *prefilter output*, and $m_{p_j}(x,y)$ as the *postfilter output* for the j^{th} channel.

A Bayesian classifier based on predicted multivariate output statistics is used in Fig. 1 to generate the classified image $c(x,y)$ from the vector output of the k filter channels. Morphological postprocessing is used to remove misclassifications near boundaries between different textures and to generate the final segmented image $i_s(x,y)$. Details on the classifier and postprocessing are provided in [8, 10].

For the system in Fig. 1, we have previously shown that the output statistics of the k channels for a given input texture t_i are approximately given by the multivariate Gaussian probability density function (pdf) $p_i(\boldsymbol{m_p}, \boldsymbol{\Theta_k})$ for a given set of filter parameters $\boldsymbol{\Theta_k}$ [7]:

$$p_i(\boldsymbol{m_p}, \boldsymbol{\Theta_k}) = \frac{1}{(2\pi)^{k/2}|\boldsymbol{C_i}|^{1/2}} \; e^{-\left(\frac{(\boldsymbol{m_p}-\boldsymbol{\mu_i})^T \boldsymbol{C_i}^{-1}(\boldsymbol{m_p}-\boldsymbol{\mu_i})}{2}\right)} \qquad (4)$$

where $\boldsymbol{m_p}$ is a vector sample of the k-dimensional postfilter-output vector, $\boldsymbol{\mu_i}$ is the mean postfilter-output vector, $\boldsymbol{C_i}$ is the covariance matrix of the postfilter outputs.

For the system in Fig 1, the predicted texture-segmentation error $\mathcal{E}_t(\mathbf{\Theta_k})$ is [7]:

$$\begin{aligned}\mathcal{E}_t(\mathbf{\Theta_k}) &\approx \sum_{\alpha=1}^{\mathcal{N}-1} \sum_{\beta=\alpha+1}^{\mathcal{N}} \frac{(\mathcal{P}_\alpha \mathcal{P}_\beta)^{1/2}}{\mathcal{N}-1} e^{-B(t_\alpha, t_\beta, \mathbf{\Theta_k})} \\ &+ \frac{1}{k}\sum_{j=1}^{k} \frac{2(\mathcal{N})(\sigma_{g_j}^2 + \sigma_{p_j}^2)}{N^2} \end{aligned} \quad (5)$$

where $\mathcal{P}_\alpha$ and $\mathcal{P}_\beta$ are *a priori* probabilities of textures t_α and t_β occurring in the image, the image dimensions are $N \times N$, $B(t_\alpha, t_\beta, \mathbf{\Theta_k}))$ is the Bhattacharyya distance [15] between textures t_α and t_β for a given set of filter channels $\mathbf{\Theta_k}$, and the term $(\sigma_{g_j}^2 + \sigma_{p_j}^2)$ approximates the combined localization effects of the Gabor prefilter and Gaussian postfilter. This total error $\mathcal{E}_t(\mathbf{\Theta_k})$ is used in our filter-design algorithm as the basis for designing the Gabor filters in Section 3.

3. Filter design algorithm

Given the predicted segmentation error $\mathcal{E}_t(\mathbf{\Theta_k})$ in (5), we proceed in this section to develop the filter-design algorithm. We first construct a collection Ψ of *individual* candidate filter channels, from which collection the set of k channels will be constructed:

$$\Psi = \{\ \theta\ \} = \{\ (u, v, \sigma_g, \sigma_p)\ \} \quad (6)$$

such that:

$$\begin{aligned}\sigma_g &\in \Sigma \\ \sigma_p &\in \{\lambda\,\sigma_g \mid \lambda \in \Lambda\} \\ (u,v) &\in \left\{ \left(\frac{\eta_1}{\sqrt{8\pi^2\sigma_g^2}}, \frac{\eta_2}{\sqrt{8\pi^2\sigma_g^2}} \right) \right\}, \end{aligned}$$

where $\eta_1 \in \{\ldots, -1, 0, 1, 2, \ldots\}$, $\eta_2 \in \{0, 1, 2, \ldots\}$, $-0.5 \leq u < 0.5$, *and* $0 \leq v < 0.5$. The set Σ contains candidate prefilter σ_g's, and Λ is a set of constants determining candidate ratios of the postfilter σ_p relative to each value of the prefilter parameter σ_g. For each value of σ_g, the center frequencies (u, v) are chosen to create an overlapping tessellation of candidate Gabor prefilters in the frequency half-plane. Finally, we note that candidate filters in Ψ are not limited to octave scalings of σ_g.

The number of possible combinations of k channels arising from the candidate filter set Ψ is prohibitive. To mitigate this problem, we use a forward-sequential filter-selection method to find the best k-channel design from the possible candidate filter-channel combinations [14]. In the forward-sequential method, the first filter channel selected is the best individual filter channel θ_1 such that $\mathcal{E}_t(\theta_1) \leq \mathcal{E}_t(\theta_\xi)$, $\forall\ \theta_\xi \in \Psi$, where $\theta_1 \in \Psi$. To proceed further, we first define the filter channel set at iteration δ of the forward-sequential algorithm as

$$\mathbf{\Theta}_\delta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_\delta \end{bmatrix} = \begin{bmatrix} u_1 & v_1 & \sigma_{g_1} & \sigma_{p_1} \\ u_2 & v_2 & \sigma_{g_2} & \sigma_{p_2} \\ \vdots & \vdots & \vdots & \vdots \\ u_\delta & v_\delta & \sigma_{g_\delta} & \sigma_{p_\delta} \end{bmatrix} \quad (7)$$

where $\delta \leq k$, and k is the number of desired filter channels in the texture-segmentation system of Fig. 1.

Using (7), subsequent steps in the forward-sequential filter-design algorithm can then be written in a recursive form. Filter parameters $\mathbf{\Theta}_\delta$ at the δth stage of the forward-sequential algorithm are then defined in terms of the filter parameters $\mathbf{\Theta}_{\delta-1}$ at stage $\delta - 1$:

$$\mathbf{\Theta}_\delta = \begin{bmatrix} \mathbf{\Theta}_{\delta-1} \\ \theta_\delta \end{bmatrix} \quad (8)$$

such that

$$\mathcal{E}_t\left(\begin{bmatrix} \mathbf{\Theta}_{\delta-1} \\ \theta_\delta \end{bmatrix} \right) \leq \mathcal{E}_t\left(\begin{bmatrix} \mathbf{\Theta}_{\delta-1} \\ \theta_\xi \end{bmatrix} \right), \quad \forall\ \theta_\xi \in \Psi$$

where $\mathbf{\Theta}_\delta$ is a function of θ_δ, and where $\mathbf{\Theta}_{\delta-1}$ is a fixed matrix established at step $\delta - 1$ of the forward-sequential procedure. The forward-sequential algorithm terminates when the desired number of filter channels k is reached (i.e., when $\delta = k$) or when a desired error $\mathcal{E}_t(\mathbf{\Theta})$ is reached.

3.1. Design algorithm

Combining the foregoing results, the procedure for designing k filter channels comprised of k Gabor prefilters and k Gaussian postfilters is:

Step 1. Construct a large collection Ψ of individual candidate filter channels using (6). A typical set of parameters for constructing Ψ would be $\Sigma = \{2, 4, 8\}$ and $\Lambda = \{1.5, 2\}$, giving possible combinations of (σ_g, σ_p) of { (2,3), (2,4), (4,6), (4,8), (8,12), (8,16) }.

Step 2. Find the best single filter-channel θ_1 such that $\mathcal{E}_t([\theta_1]) \leq \mathcal{E}_t([\theta_\xi])\ \forall\ \theta_\xi \in \Psi$.

Step 3. Search for subsequent filters using the forward-sequential algorithm in (8), terminating when the desired number of filter channels k is reached. Alternatively, terminate the algorithm when the predicted segmentation error $\mathcal{E}_t(\mathbf{\Theta})$ reaches some desired level.

4. Results

Fig. 2 shows results of our algorithm for an 8-texture image. The 256×256 pixel 8-bit gray-scale input image

(a) (b) (c)

Figure 2: Results for $\Sigma = \{2, 3, 4.5\}, \Lambda = \{1.5\}$. (a) Input composite 256×256 image comprised of Brodatz textures d77, d84, d55, d17, d24, d21, d57, d68. (b) Two channel segmentation, error=0.09, $\mathbf{\Theta}_2 = [(0.19, 0, 4.5, 6.75), (0.17, 0.125, 4.5, 6.75)]^T$. (c) Six channel segmentation, error=0.08, $\mathbf{\Theta}_6 = [(0.19, 0, 4.5, 6.75), (0.17, 0.125, 4.5, 6.75), (-0.08, 0.13, 4.5, 6.75), (0.11, 0, 4.5, 6.75), (-0.3, 0.25, 2, 3), (0, 0.19, 3, 4.5)]^T$.

in Fig. 2(a) consists of eight Brodatz textures [16]. The segmented image $i_s(x, y)$ using two filter channels is shown in Fig. 2(b) for the image processing system in Fig. 1. Finally, a segmentation using six filter channels is shown in Fig. 2(c) with only 8% of the pixels in the image being misclassified. Using 2-6 Gabor filters, we achieve effective segmentation results for 8 textures. By comparison, Jain and Farrokhnia obtained similar segmentation results using 13 filters [2] and Randen and Husøy also achieved similar results using 13 to 40 filters [17] in images containing only 5 textures. Further results for a variety of synthetic and natural textures are given in Figs. 3 and 4.

5. References

[1] A. C. Bovik, "Analysis of multichannel narrow-band filters for image texture segmentation," *IEEE Trans. Signal Processing*, vol. 39, no. 9, pp. 2025–2043, Sept. 1991.

[2] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," *Pattern Recognition*, vol. 23, no. 12, pp. 1167–1186, Dec. 1991.

[3] D. F. Dunn and W. E. Higgins, "Optimal Gabor filters for texture segmentation," *IEEE Trans. Image Proc.*, vol. 4, no. 7, pp. 947–964, July 1995.

[4] D. Dunn, W. Higgins, and J. Wakeley, "Texture segmentation using 2-D Gabor elementary functions," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, no. 2, pp. 130–149, Feb. 1994.

[5] T. P. Weldon, W. E. Higgins, and D. F. Dunn, "Gabor filter design for multiple texture segmentation," *Optical Eng.*, vol. 35, no. 10, pp. 2852–2863, Oct. 1996.

[6] T. P. Weldon, W. E. Higgins, and D. F. Dunn, "Efficient Gabor filter design for texture segmentation," *Pattern Recognition*, vol. 29, no. 12, pp. 2005–2015, Dec. 1996.

[7] T. Weldon and W. Higgins, "Design of multiple Gabor filters for texture segmentation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. IV, (Atlanta, GA), pp. 2245–2248, 7-10 May 1996.

[8] T. P. Weldon and W. E. Higgins, "Integrated approach to texture segmentation using multiple Gabor filters," in *Proc. IEEE Int. Conf. on Image Processing*, vol. III, (Lausanne, Switzerland), pp. 955–958, Sept. 1996.

[9] T. P. Weldon, W. E. Higgins, and D. F. Dunn, "Efficient Gabor filter design using Rician output statistics," in *Proc. IEEE Int. Symp. Circuits, Systems*, vol. 3, (London, England), pp. 25–28, 30 May - 2 June 1994.

[10] T. P. Weldon, *Multiresolution Design of Multiple Gabor Filters for Texture Segmentation.* PhD thesis, The Pennsylvania State University, 1995.

[11] J. G. Daugman, "Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, no. 7, pp. 1169–1179, July 1988.

[12] T. Randen and J. H. Husøy, "Novel approaches to multichannel filtering for image texture segmentation," in *Proc. SPIE Visual Comm. Image Processing 1994*, vol. 2094, pp. 626–636, 1994.

[13] T. Chang and C. C. J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Trans. Image Proc.*, vol. 2, no. 4, pp. 429–441, Oct. 1993.

[14] L. Kanal, "Patterns in pattern recognition: 1968-1974," *IEEE Trans. Inform. Theory*, vol. 20, no. 6, pp. 697–722, Nov. 1974.

[15] W. K. Pratt, *Digital Image Processing.* John Wiley and Sons, second ed., 1991.

[16] P. Brodatz, *Textures: A Photographic Album for Artists and Designers.* New York, NY: Dover, 1966.

[17] T. Randen and J. H. Husøy, "Multichannel filtering for image texture segmentation," *Optical Eng.*, vol. 33, no. 8, pp. 2617–2625, Aug. 1994.

(a) (b) (c)

Figure 3: Results for $\Sigma = \{2.5, 5, 10, 20\}$, $\Lambda = \{1.7\}$. (a) Input composite 256×256 image comprised of textures "noise," d29, d9, "+," d84. (b) Two channel segmentation, error=0.07, $\mathbf{\Theta}_2 = [(-0.31, 0.31, 5, 8.5), (0.14, 0, 5, 8.5)]^T$. (c) Four channel segmentation, error=0.05, $\mathbf{\Theta}_6 = [(-0.31, 0.31, 5, 8.5), (0.14, 0, 5, 8.5), (0.09, 0.05, 5, 8.5), (-0.41, 0.34, 2.5, 4.25)]^T$.

(a) (b) (c)

Figure 4: Results for $\Sigma = \{2.5, 5, 10, 20\}$, $\Lambda = \{1.7\}$. (a) Input composite 256×256 image comprised of textures d12, d17, d55, d32, d29. (b) Two channel segmentation, error=0.11, $\mathbf{\Theta}_2 = [(0.08, 0.13, 5, 8.5), (0, 0.20, 5, 8.5)]^T$. (c) Four channel segmentation, error=0.07, $\mathbf{\Theta}_6 = [(0.08, 0.13, 5, 8.5), (0, 0.20, 5, 8.5), (-0.02, 0.06, 5, 8.5), (0.19, 0, 2.5, 4.25)]^T$.