Homework #4: Face Recognition: Scattered Wavelet Network

Due April 19th, 2016

Professor Davi Geiger.

The goal of this assignment is to develop a face recognition machine. It is based on the work of Bruna and Mallat, 2012. The idea is to build a convolution-like Network, with various layers (typically 2 or 3 layers). At each layer an "image-like" is convolved with a low pass filter ϕ_σ (a blur-like filter at scale σ) and down sampled (reducing the size). We refer to "image-like" to structures that have the same pixels as the image and are non-negative quantities at each pixel. The "image-like" vary from layer to layer via a prescribed procedure as we describe next.

At the zero layer, we start the input image, I, i.e., $I^{\mathbf{0}}=I$. At the next layer, layer-1, the image I is convolved with a bank of $\boldsymbol{m}$ wavelets at different directions and different scales, represented by $\psi_\lambda * I$, where $\lambda_i=(\sigma_i, \theta_i)$ represents a given scale and orientation, ie.., $i=1,\ldots\boldsymbol{m}$. The "images-like" at this layer becomes the magnitude of each convolution, $I^{\mathbf{1}}_i=|\psi_{\lambda i} * I^{\mathbf{0}}|$. We have as many "images-like" at this layer as orientations and scales of wavelets. At the next layer, layer-2, we have $I^{\mathbf{2}}_k=|\psi_{\lambda j} * I^{\mathbf{1}}_i|$, $k=(i,j)$. Again, the saved vector is the down sampled of: each "image-like" convolved with a low pass filter, i.e., the down sample of $v^{\mathbf{L}}_k = \phi_\sigma * I^{\mathbf{L}}_k$. The final representation, what is saved, is a set of layer vectors $v^{\mathbf{L}}_k$ downsampled. This representation is created to be invariant to small deformation made to each input image. So if two images are similar, up to small deformations on the grid, then the final output of the scattered network will produce vectors that are close to each other in Euclidean distance.

Let us develop and test these ideas on face recognition.

Morlet Wavelets: You have done all of this already. I am suggesting to use just Scales $\boldsymbol{\sigma} = 3, 5$, **Angles**, $\boldsymbol{\theta} = 0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}$

SCATTERED WAVELET NETWORK "SCHEME"

	Layer 0	**Layer 1**	**Layer 2**
Wavelet Modulus	$\boldsymbol{I(u)}$	$\left\|\boldsymbol{\psi_{\lambda_i} * I(u)}\right\|$ $\boldsymbol{i = 1, \ldots, 8;}$	$\left\|\boldsymbol{\psi_{\lambda_j} * \left\|\psi_{\lambda_i} * I(u)\right\|}\right\|$ $\boldsymbol{i = 1, \ldots, 8;\ j = 1, \ldots, 4}$
GaussianConvolution	$\boldsymbol{G_{\sigma=5} * I(u)}$	$\boldsymbol{G_{\sigma=5} * \left\|\psi_{\lambda_i} * I(u)\right\|}$	$\boldsymbol{G_{\sigma=5} * \left\|\psi_{\lambda_j} * \left\|\psi_{\lambda_i} * I(u)\right\|\right\|}$
Downsampling			
OUTPUT:	$\boldsymbol{S^0_{k=3}(I)}$	$\boldsymbol{S^1_{k=3}\left(\left\|\psi_{\lambda_i} * I(u)\right\|\right)}$	$\boldsymbol{S^2_{k=3}\left(\left\|\psi_{\lambda_j} * \left\|\psi_{\lambda_i} * I(u)\right\|\right\|\right)}$

MORE PRECISELY

ZERO LAYER

Convolve the image with a Gaussian Blur (ϕ_σ=$\boldsymbol{G}_{\sigma=5}$**).**

$$\boldsymbol{S^0}(\boldsymbol{I}) = \boldsymbol{G}_{\sigma=5} * \boldsymbol{I}(\boldsymbol{u})$$

We then downsample the output by a factor 2^3 along the x and y axis. Let us also consider downsampling to one pixel (a factor 2^5) . Let us refer the downsample by 2^k as $\boldsymbol{S^0_{k=3}}(\boldsymbol{I})$ or $\boldsymbol{S^0_{k=5}}(\boldsymbol{I})$ If the image $\boldsymbol{I}(\boldsymbol{u})$ is of size 32×32, then $\boldsymbol{S^0_{k=3}}(\boldsymbol{I})$ is of size 4×4, i.e., it contains 16 parameters (the values of $\boldsymbol{S^0_{k=3}}(\boldsymbol{I})$ at each of the 4×4 locations). Similarly $\boldsymbol{S^0_{k=5}}(\boldsymbol{I})$ contains 1 parameter. One can refer to the elements $[S^0_{k=3}(I)]_{pq}$, $p, q = 1, \dots, 4$.

FIRST LAYER

For each image and each λ_i, perform a convolution with the real and imaginary part of the wavelet and produce the results,

$$\boldsymbol{W}_{\lambda_i}\boldsymbol{I}(\boldsymbol{u}) = \boldsymbol{\psi}_{\lambda_i} * \boldsymbol{I}(\boldsymbol{u}) \qquad \boldsymbol{i = 1, \dots, 8}$$

Scales $\boldsymbol{\sigma} = 3, 5,$ **Angles,** $\boldsymbol{\theta} = 0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}$

or

$$\boldsymbol{W}^{\boldsymbol{Real}}_{\lambda_i}\boldsymbol{I}(\boldsymbol{u}) = \boldsymbol{\psi}^{\boldsymbol{Real}}_{\lambda_i} * \boldsymbol{I}(\boldsymbol{u}) = \sum_{\boldsymbol{x'}}\sum_{\boldsymbol{y'}} \boldsymbol{I}(\boldsymbol{x'}, \boldsymbol{y'})\, \boldsymbol{\psi}^{\boldsymbol{Real}}_{\lambda_i}(\boldsymbol{x'} - \boldsymbol{x}, \boldsymbol{y'} - \boldsymbol{y}) \qquad \boldsymbol{i = 1, \dots, 12}$$

$$\boldsymbol{W}^{\boldsymbol{Im}}_{\lambda_i}\boldsymbol{I}(\boldsymbol{u}) = \boldsymbol{\psi}^{\boldsymbol{Im}}_{\lambda_i} * \boldsymbol{I}(\boldsymbol{u}) = \sum_{\boldsymbol{x'}}\sum_{\boldsymbol{y'}} \boldsymbol{I}(\boldsymbol{x'}, \boldsymbol{y'})\, \boldsymbol{\psi}^{\boldsymbol{Im}}_{\lambda_i}(\boldsymbol{x'} - \boldsymbol{x}, \boldsymbol{y'} - \boldsymbol{y}) \qquad \boldsymbol{i = 1, \dots, 12}$$

Compute the magnitude of the wavelet transform, $\left|\boldsymbol{W}_{\lambda_i}\boldsymbol{I}(\boldsymbol{u})\right|$, where

$$\left|\boldsymbol{W}_{\lambda_i}\boldsymbol{I}(\boldsymbol{u})\right| = \sqrt{\boldsymbol{W}_{\lambda_i}\boldsymbol{I}(\boldsymbol{u}) \cdot \text{conjugate}(\boldsymbol{W}_{\lambda_i}\boldsymbol{I}(\boldsymbol{u}))} = \sqrt{\left(\boldsymbol{W}^{\boldsymbol{Real}}_{\lambda_i}\boldsymbol{I}(\boldsymbol{u})\right)^{\boldsymbol{2}} + \left(\boldsymbol{W}^{\boldsymbol{Im}}_{\lambda_i}\boldsymbol{I}(\boldsymbol{u})\right)^{\boldsymbol{2}}}$$

and in order to down sample it, first convolve it with the Gaussian $\boldsymbol{G}_{\sigma=5}$

$$\boldsymbol{S}^{\boldsymbol{1}}_{\lambda_i}(\boldsymbol{I}) = \boldsymbol{G}_{\sigma=5} * \left|\boldsymbol{W}_{\lambda_i}\boldsymbol{I}(\boldsymbol{u})\right|$$

Finally, we also down sample the output by a factor 2^3 along the x and y axis, to make $\boldsymbol{S}^{\boldsymbol{1}}_{\lambda_i,k=3}(\boldsymbol{I})$. For the image $\boldsymbol{I}$ of size 32×32, $\boldsymbol{S}^{\boldsymbol{1}}_{\lambda_i,k=3}(\boldsymbol{I})$ is of size 4×4 for each of the eight

λ_i. $\boldsymbol{S}^{\mathbf{1}}_{\lambda_i,k=3}(\boldsymbol{I})$ can be thought as a vector of size 4×4 =16 where each entry value is the quantity $\boldsymbol{S}^{\mathbf{1}}_{\lambda_i,k=3}(\boldsymbol{I})$ at the corresponding location in the 4×4 grid**,** i.e.,

$$\left[S^1_{\lambda_i}(I)\right]_{p,q} \quad p,q = 1,\dots,4 \qquad or \qquad \left(S^1_{\lambda_i}(I)\right)_l \quad l = 1,\dots.,16$$

We will be using the vector notation $\left(S^1_{\lambda_i}(I)\right)_l$. All together, we have 8 vector image-like structures $\{\boldsymbol{S}^{\mathbf{1}}_{\lambda_i,k=3}(\boldsymbol{I}), i = 1,\dots 8\}$ in layer 1, each of size 16. Similarly we study $\{\boldsymbol{S}^{\mathbf{1}}_{\lambda_i,k=5}(\boldsymbol{I}), i = 1,\dots 8\}$ where$\left(S^1_{\lambda_i}(I)\right)_l \quad l = 1$.

PROBLEM 1: PROCESS THE DATABASE BY THE SCATTERED WAVELET NETWORK

Let us work first with just these two layers, $\vec{S}(I) = \{\boldsymbol{S}^{\mathbf{0}}(\boldsymbol{I}), \boldsymbol{S}^{\mathbf{1}}_{\lambda_i}(\boldsymbol{I}), \boldsymbol{i} = \mathbf{1},\dots,\mathbf{8}\}$, where $\boldsymbol{S}^{\mathbf{0}}(\boldsymbol{I})$ is of size 16 (or 1) and the 8 scattered $\boldsymbol{S}^{\mathbf{1}}_{\lambda_i}(\boldsymbol{I})$ are also each of size 16 (or 1). One can think of $\vec{S}(I)$ as 9 vectors of size 16 (or size 1, for k=5) with entries given by the scattered network described above. Alternatively, we can write $\vec{S}(I)$ as a long vector of size $\mathbf{144} = (1+8) \times 16$, a concatenation of the vectors $\left(S^1_{\lambda_i}(I)\right)_l$ into one long vector. Alternatively, for k=5. we can write $\vec{S}(I)$ as a long vector of size $\mathbf{9} = (1+8)$, a concatenation of the vectors $\left(S^1_{\lambda_i}(I)\right)_{l=1}$ into one long vector. This is our new image representation. Every image of a face on the database becomes a vector $\vec{S}(I)$ of size $\mathbf{144}$ (or size **9**). Compute them for the image database of faces.

EXTRA: OPTIONAL. IMPROVING WITH THE SECOND LAYER

Reapply the same concept of convolution with wavelets followed by a magnitude measure, followed by averaging with Gaussian, i.e.,

$$\boldsymbol{S}^{\mathbf{2}}_{\lambda_j,\lambda_i}(\boldsymbol{u}) = \boldsymbol{G}_{\sigma=6} * \left|\boldsymbol{\psi}_{\lambda_j} * \left|\boldsymbol{\psi}_{\lambda_i} * \boldsymbol{I}(\boldsymbol{u})\right|\right|$$

But now only use the large scale and four angles $\lambda_j = (\sigma = 5, \theta_j)$, i.e., there are only 4 different values for the parameter set λ_j. Finally, we also downsample the output by a factor 2^3 along the x and y axis (or downsample to only one value). In total we have new $\mathbf{32} = 4 \times 8$ different $\boldsymbol{S}^{\mathbf{2}}_{\lambda_j,\lambda_i}(\boldsymbol{u})$ downsampled "scattered images" in layer 2. Each $\boldsymbol{S}^{\mathbf{2}}_{\lambda_j,\lambda_i}(\boldsymbol{u})$ downsampled to 4 x 4 has length 16, so this layer becomes a long vector of size ***512*** = 32 x 16. If we downsample to 1 element, it becomes of size $\mathbf{32.}$ Extend Problem 1, now adding the second Layer to the representation and so create the much longer vector.