Progress Report
18-30 September 2012

Amir Hesam Salavati
E-mail: hesam.salavati@epfl.ch

Supervisor: Prof. Amin Shokrollahi
E-mail: amin.shokrollahi@epfl.ch

Algorithmics Laboratory (ALGO)
Ecole Polytechnique Federale de Lausanne (EPFL)

October 4, 2012

1 Summary

In the pas two weeks, Mr. Amin Karbasi and I have been busy preparing our work for the ICML 2013 conference. For this purpose, we have witten a draft (which can be found separately in the ERC log) as well as performing new simulations over real datasets. The paper is based on the (rejected) NIPS 2012 draft, with a few modifications, and it now contains simulations over a dataset of spoken English words. The results were very promising and have encouraged us to proceed in this direction.

On a related topic, we were working to perform simulations over a dataset of natural images. We have played around with different models that suits our error correcting algorithm and at the moment, have a very promising one, explained in more details in what follows. In this report, we will also discuss some of the models that have failed and the reason behind their failure.

Currently, we are interested in memorizing and denoising images. But we also have an eye on extending our algorithm to image classification. something which might require some radical changes in the recall (classification) algorithm.

2 Research on datasets of natural images

To apply our neural associative algorithm to any dataset, we need inputs to fulfill the following properties:

1. They (almost) form a subspace so that the dual vectors can be learned using our neural learning algorithm.
2. The dual vectors are rather sparse. Not all datasets lead to sparse dual vectors.

For synthetic datasets, inputs directly satisfy the above properties (we make them in such a way!). However, for natural stimuli, it is not usually the case and one has to find a *transform* to map the input stimuli to another space where the new transformed vectors (almost) form a subspace. The mapping can be non-linear. Depedning on the required task, the mapping can also be irreversible, i.e. one can not get back to the original stimuli from the mapped sequences. For dataset of spoken English words, one notices that a simple FFT is sufficient as a mapping, as shown in previous reports and our ICML paper [1].

For dataset of natural images, we have been playing around with different mappings. In the previous report, we dicussed a mapping that is irreversible, i.e. one can not reconstruct the input image from the features. This model is shown in Figure 1.

In this report, we focus on a mapping that is reversible up to some point. More specifically, we would like to find a mapping that allows us to roughly reconstruct the input image from its features. Furthermore, we are interested in finding features in a way that they satisfy the above two properties, i.e. they form a subspace and allow sparse dual vectors. After playing around with different models, we have found the one shown in Figure 2 very effective in this regard.

In words, the proposed model is composed of the following stages:

1. We apply the Gabor wavelet to the image by computing the 2D convolution of 16 Gabor filters and the image. Each of Gabor filters is responsible for oriented edge detection, with orientations varying from 0 to 180 degrees in 16 steps.

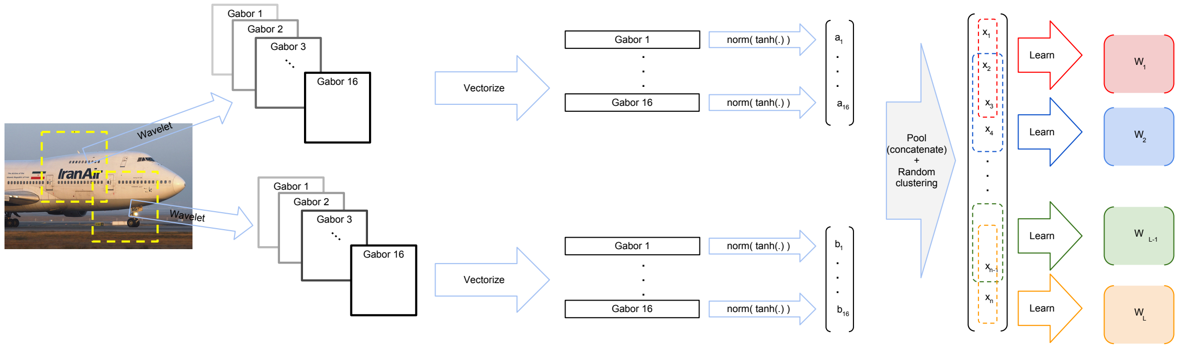

Figure 1: An exmaple of irreversible feature mapping. The mapping is irreversible because of the tanh, |.| and *norm* functions.

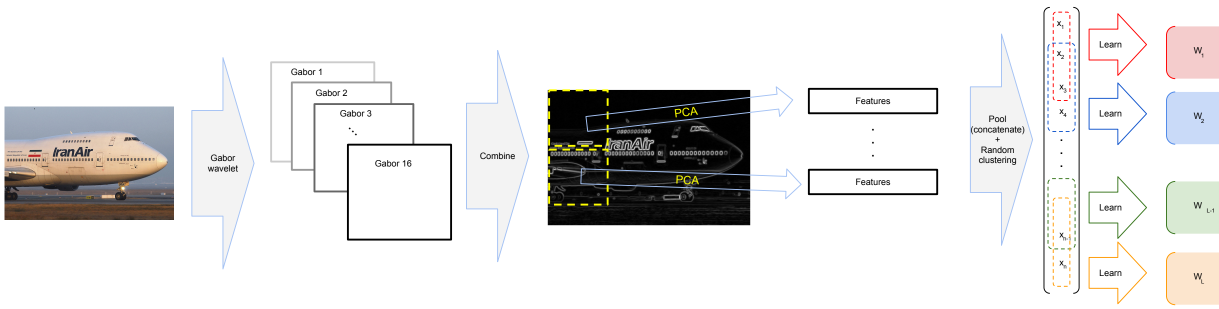

Figure 2: An exmaple of rreversible feature mapping. The mapping is reversible upto the wavelet reconstruction of the image (in black and white).

2. We then divide the newly constructed image into smaller ***non-overlapping*** patches. The patches should be non-overlapping due to the fact that we require the mapping to be reversible. More specifically, if the patches are overlapping, when we apply the PCA or SVd in the next stage, each pixel will be part of multiple mappings, which makes it difficult in reconstructing the original image from its features, i.e. it would be difficult to decide the calue of each pixel when it belongs to more than one cluster.

3. Next, we apply PCA or SVd to each patch separately. The resulting scores/coefficients with respect to these coordinates (i.e. principle components) will be used as the input for the learning layer.

4. We concatenate (pool) all the scores/coefficients of the PCA/SVD from the previous layer to form a very large vector of length n. We then divide these entries into L random ***overlapping*** clusters.

5. The usual neural learning algorithm is then applied to each cluster and the result is stored as the weight matrices $W_1, \ldots, W_L$.

This model has been tested in practice and it seems to work. Therefore, we might be able to use this model as a base to denoise images.

2.1 Other variants

A variant to the above model that could work comes from removing step 2. More specifically, in step 2 of the above procedure we combine the 16 different filtered outputs to get a single (filtered) image in the end. We could ignore this step and keep all 16 images as the input to the next stage, where we apply PCA or SVD. The resulting output is in the same spirit of the above model, except for its larger length (16 times larger).

Another possible approach is to divide each input image into smaller patches and then applying Gabor wavelet to each of them. Note that it is better for the patches to be ***non-overlapping***, because we would have difficulties in combining them later otherwise, i.e. when we are going to reconstruct the new image after applying the wavelet transform, we would have problems in combining various pixels if the patches are overlapping. Figure 3 illustrates this model.

Note that in all cases, the patching before applying PCA or SVD is non-overlapping due to the reasons explained before.

3 Image classification and our model

We have also thought about using our model for image classification. For this purpose, we should learn L dual matrices for each class. Furthermore, for each matrix $W_i^{(j)}$ corresponding to cluster i of class j, we must have the following properties:

1. For each pattern x in the test set of class j, T_j, we must have $W_i^{(j)} \cdot x = 0$, for all $i = \{1, \dots, L\}$.

2. For each pattern x' in the training or test sets of class j', we must have $W_i^{(j)} \cdot x \neq 0$.

We have tested the above properties for the STL-10 dataset and unfortunately, the first property above does not hold. Therefore, we must find a more appropriate model for in order to use our network in image classification.

4 Conclusions and future works

So far, we have found a model that seems to work for denoising purposes (as shown in Figure 2). In the following weeks, we are going to apply this model to denoise natural images from real datasets.

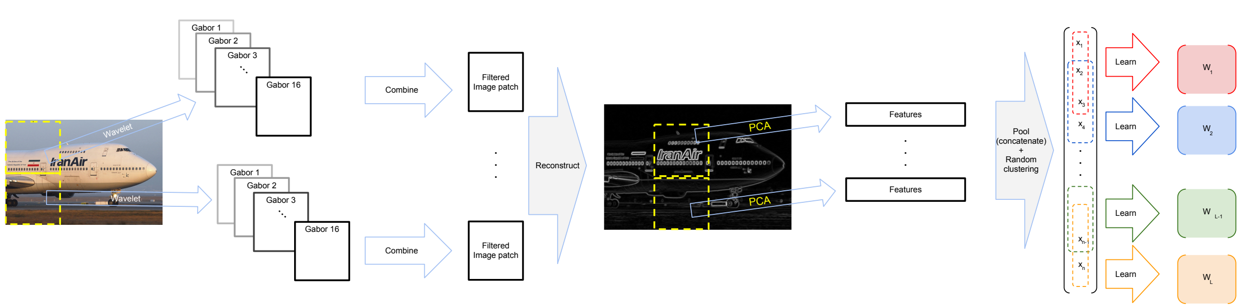

Figure 3: An example of reversible feature mapping. The mapping is reversible upto the wavelet reconstruction of the image (in black and white).

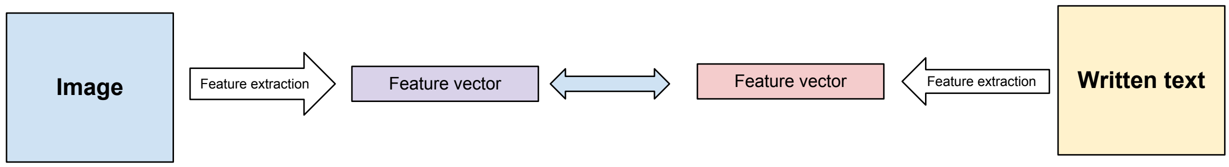

Figure 4: Feature extraction and associative memory

Previously, we have also shown that there is a model which can become handy in denoising *image features* (rather than the image itself) since the mapping is irreversible (see Figure 1). This model can be used in associating for instance the features of an image with features of, for instance, written words. Therefore, we might have a model like the one shown in Figure 4, which can be used in designing associative memories.

Finally, we should find a proper model for classification of natural images.

References

[1] A. Karbasi, A. H. Salavati, A. Shokrollahi, *Iterative learning and denoising in convolutional neural associative memories*, Submitted to ICML 2013.