ADAPT
IST-2001-37173
Artificial Development Approach to Presence Technologies

Deliverable Item 5.3
Initial Implementation of the Integration Model

Delivery Date: May 21, 2004
Classification: Public
Responsible Person: Harri Valpola
Partners Contributed: ALL

Short Description: We have implemented a model which is ment for extracting coherent representations from multi-modal data. The model consists of a hierarchy of interconnected units. The individual units were discussed in D5.2 but their design has been further simplified and improved. In this document, we describe initial experiments with the model. We report here experiments which verify that the model works in principle. Experiments with more realistic multi-modal data will be reported in the forthcoming D5.4.

Project funded by the European Community under the "Information Society Technologies" Programme (1998–2002)

1 Introduction

The feature extraction model we are developing is a hierarchy of basic units, feature maps, which were reported in D5.2. The goal of the model is to extract a coherent, behaviourally meaningful representation of the sensory input. The term "behavioural meaningful" refers to the fact that the representation is supposed to support motor control, prediction of future rewards, etc. The level of abstraction in the representation is supposed to increase towards the higher levels in the hierarchy as explained in D5.1.

While D5.2 emphasized invariant features, we have now slightly shifted our thinking and emphasize more contextual coherence. Temporal invariance can be regarded as a special case of contextual coherence, where context is taken to be temporal. There are other types of contextual coherence and the experiments we report here demonstrate spatial coherence in visual processing. In the future, we expect to be able to use this principle to develop coherent representations in multi-modal data.

As in the model reported in D5.2, we still have a division between bottom-up and contextual inputs. In D5.2, contextual inputs were assumed to include lateral and top-down inputs. Now we also assume temporally delayed inputs to be part of contextual inputs. Bottom-up inputs are assumed to be primary in the sense that the extracted features, activations, will reflect the bottom-up input activations. The role of contextual inputs is modulatory. They decide what kind of features the feature map will learn to extract.

The structure of the basic unit has been simplified in several ways since D5.2. Some of the features of the basic unit were designed to make the model scalable and we expect those features to play a role in the future when larger models will be implemented. At the present stage, we concentrate mainly on studying how context can guide the development of representation in relatively small models and we therefore decided to resort to simple solutions.

The development of the feature extraction model has been guided by theoretical, biological and practical considerations. Discussion about the motivations is nevertheless mostly postponed to D5.4 which is a "report". The present deliverable is a "prototype" and therefore mainly the implementation and initial results are reported here. This work has also been discussed in [3] together with more motivation and background. Related theoretical background can be found in [1, 2].

2 Hierarchy of feature extraction units

Our model is closely related to the hierarchy of feature extractors used in slow feature analysis (SFA) [4]. Figure 1 depicts the structure of the model. The goal of the model is to extract features which change slowly in time. The model consists of a hierarchy of units which each process part of the input data and extract nonlinear invariant features. The higher levels of the hierarchy integrate features from neighbouring units into more and more invariant features.

Each units processes the inputs in two stages. At the first stage, nonlinear features are computed from the inputs and the resulting features are sphered. Sphering refers to a decorrelation and normalisation process where the variance structure of the data is removed. In practice, sphering is implemented by principal component analysis (PCA) followed by normalisation of the vari-

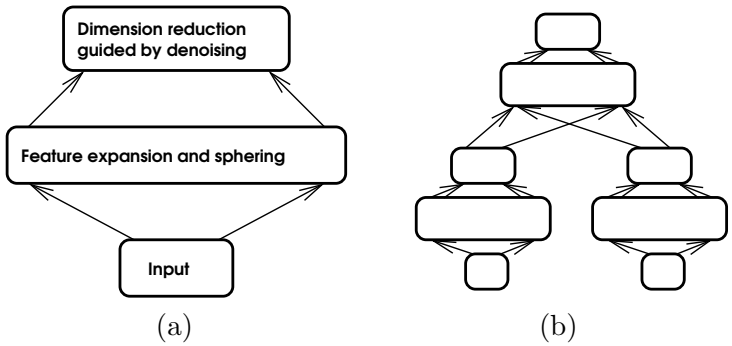

Figure 1: Structure of the model in slow feature analysis. The model consists of a hierarchy of separate units. Each unit (a) consists of two processing stages. At the first stage, a set of nonlinear features are computed from the inputs and the resulting features are sphered, i.e. the variance structure is removed. After sphering, the variance of data is uniform in every direction. At the second stage, the data is low-pass filtered. This renders the variance of those directions highest where the signal changes slowest. The slow features can then be identified by linear dimension reduction (PCA). A hierarchy of units (b) can develop a representation with increasing levels of abstraction.

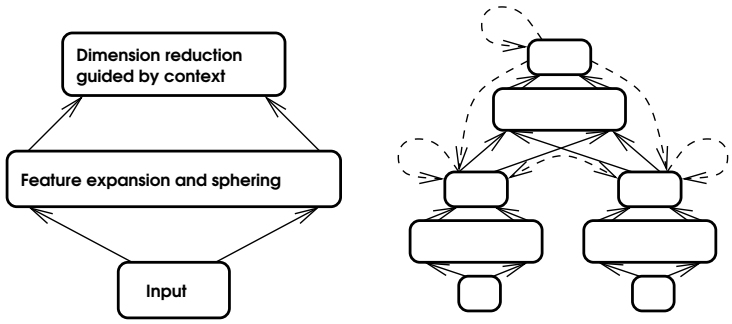

Figure 2: Structure of our model. The main difference compared to slow feature analysis is the criterion for dimension reduction. We augment the sphered inputs with a context before dimension reduction. The context has small weights compared to the sphered inputs, but the result is nevertheless that the augmented data is no longer exactly spherical. Linear dimension reduction can then identify those directons where the bottom-up inputs have a correlation with the context. If delayed versions of the output of each unit are used as the context for the same unit, this is essentially equivalent to slow feature analysis. In our model, context also includes the outputs of different units at the same or higher level in the hierarchy. This means that each unit is trying to extract features which are coherent not only temporally but also in light of the information represented by the other units.

ances of the components. The next stage identifies those projections of the sphered data that change most slowly. In [4], this was achieved by high-pass filtering the data and then finding the minor components, i.e. those directions where the remaining variance is lowest.

Our work in source separation [1] has led us to reinterpret the second processing stage in SFA. It turns out that the procedure is equivalent to low-pass filtering the sphered features and then applying ordinary PCA. According to this interpretation, low-pass filtering is a denoising procedure. This interpretation suggest several obvious extensions [1] and has also led to the present model, shown in Fig. 2. The only significant difference with SFA is that denoising is more general. Before the second dimension reduction stage, the sphered inputs are augmented by a context. The goal of this augmentation is to perturb the variance structure of the sphered data. Even the slighest perturbation is enough to render the variance of the desired directions higher. This means that the weights of the context in the augmented vector can be far smaller than the weights of the sphered features of bottom-up inputs. This way the output activations of a unit reflect the bottom-up input activations but learning is guided by the context.

3 Implementation

At the present stage, we used Matlab to implement the model. Learning uses batch model, i.e. the whole data is processed as one chunk. The main algorithmic component of the model is PCA which is used both in the sphering and dimension reduction stage. As there are online versions of PCA, it should be possible to come up with an online-mode algorithm later on.

Note that PCA does not assume any topology in the inputs or outputs. We expect that later versions of the model may rely on topological feature map as explained in D5.2. In the present model, we split the data to be processed by discrete units with no internal topology. The strict boundaries between the units may be faded away later on if we decide to use a large topographically organised map where decorrelation and normalisation are implemented by local competition and where a smooth transition between units is implemented by restricted bottom-up input arbors.

Besides the structure of the model (number of units, extracted features in each unit, and connectivity between the units), the only free parameter in the present model was the ratio between the bottom-up and contextual weights in the augmented vectors before dimension reduction. In the experiments we have so far used an arbitrarily chosen a ratio of 90 % weights for bottom-up weights and 10 % for contextual weights. The ratio is measured from the sum of absolute values of the weights. Note that since the context included the outputs of potentially very many units, contextual inputs can easily outnumber the bottom-up inputs.

4 First experiments

In [3], first experiments with the model are reported. The idea is to use to a hierarchical model later on for multi-modal data, but now we only showed that the idea of context-guided denoising works in principle. The inputs were 21×21 windows taken from preprocessed raw images. This window was split into a 3×3 sub-windows, each of size 7×7 pixels. Each sub-window was processed by one feature extraction unit and the context for each unit was the output of the eight remaining units. The preprocessed image is shown on the top left of Fig. 3 together with

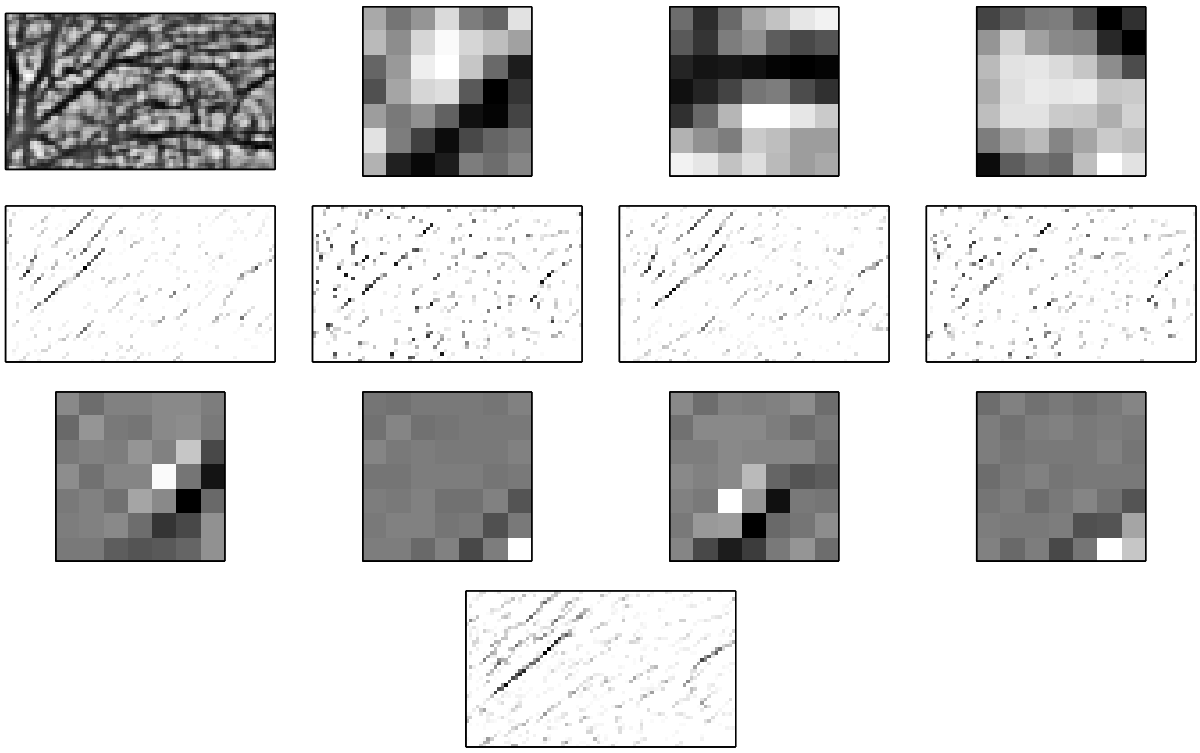

Figure 3: Top row: gray-scale image (left) and three 7×7 patches. Second and third row: activations at each image location and receptive fields of four neurons from the feature expansion layer. Bottom: output activations of a neuron pooling from the four features.

three out of 5,220 sub-windows on the top right.

Our goal was to show that the information processed by each unit can guide the extraction of information in the other units. Although we only used inputs from visual modality, conceptually the situation does not differ from having inputs from different modalities. Since each unit was now processing visual input, the representation of each unit is presumably more easily related to the representation of other units than in multi-modal case. We expect that a hierarchy of units can solve this. Now the hierarchy was flat, i.e. there were no higher level units.

The extracted features turned out to have a response that is somewhat invariant to translation. An example of such complex-cell-like feature which integrates four elementary edge detector features is shown in Fig. 3. The output feature is cleaner and more invariant than the constituent features at the feature expansion stage.

Note that most neurons at the feature expansion stage had even noisier outputs than the ones shown in the figure but they were not used as much to build the output features. The experiment thus shows that even with very limited data set (usually several natural images are used) and without using temporal context, it is possible to develop meaningful features by contextual guidance. Very weak contextual guidance is sufficient when the feature expansion stage provides sphered outputs.

References

[1] Jaakko Särelä and Harri Valpola. Denoising source separation. *Journal of Machine Learning Research*, 2004. Submitted. Available at Cogprints `http://cogprints.ecs.soton.ac.uk/archive/00003493/`.

[2] H. Valpola and J. Särelä. Accurate, fast and stable denoising source separation algorithms. In *Proceedings of the 5th International Conference on Independent Component Analysis and Blind Signal Separation (ICA 2004)*, Granada, Spain, 2004. In press. Available at Cogprints `http://ailab.ch/people/valpola/papers/ICA04.pdf`.

[3] Harri Valpola. Behaviourally meaningful representations from normalisation and context-guided denoising. Technical report, Artificial Intelligence Laboratory, University of Zurich, 2004. Available at Cogprints `http://cogprints.ecs.soton.ac.uk/archive/00003633/`.

[4] L. Wiskott and T. Sejnowski. Slow feature analysis: Unsupervised learning of invariances. *Neural computation*, 14:715–770, 2002.