Feature Processing for Automatic Anatomical Landmark Detection Using Reservoir Networks

Benjamin Roeschies, Susanne Winter

Institut für Neuroinformatik, Ruhr-Universität Bochum, 44780 Bochum, Germany
Benjamin.Roeschies@neuroinformatik.rub.de,
Susanne.Winter@neuroinformatik.rub.de

Abstract. We present an approach to the combination of an arbitrary number of image features to produce more sophisticated features for anatomical landmark detection. The combination was done using reservoir networks. The results were compared to Gabor wavelet features on single point detection in 2D slices of MR image data to show behavior and potential quality of the combined features.

1 Introduction

The automatic detection of anatomical landmarks in medical data is a challenging task [1,2]. Many computer assisted procedures in modern medicine use landmarks e. g. for intraoperative registration in navigated surgery. They can also serve as starting points in image segmentation for automatic diagnosis. Recently, a landmark detection techniqe for CT data has been proposed [3], that is based on a bunch graph matching [4] with Gabor wavelet features [5], which are used frequently for recognition tasks in computer vision. This method, however, failed in our experiments on MR image data, which is probably because the Gabor features are not adequate in this case.

The graph matching algorithm performs a global search for several landmarks at once by combining them in a pre-defined topology. This initial global search leads to a starting position for each landmark, which should be already close to the correct position. Thus, features are needed that produce local similarity maxima at the correct target positions within a limited area. A global similarity maximum for the correct point is not a requirement, because a global search for possible regions in which certain points are located is solved by the consideration of topology during the bunch graph matching.

In our approach we started with the extraction of features from an input image. Afterwards, these features served as input for a two-dimensional recurrent neural network as a processing unit. This network was composed of 4 identical neuron layers. Each layer processed the input in a different direction. One point was processed at a time by feeding its scalar feature values – i. e. the elements of the feature tensors at that given point – into the network. This way of image processing using recurrent neural networks has been introduced recently [6]. In contrast to that approach, however, we use a different network type, introduced

as Echo State Networks [7] or Liquid State Machines [8]. The basic idea of this network concept is to use a comparably large recurrent network and randomly initialize the interconnection weights, which remain untrained. This principle is referred to as reservoir computing.

For comparison of two points or landmarks, a similarity measure was then applied to the network's outputs corresponding to these points.

2 Materials and Methods

2.1 Image Data

For our experiments we used T2 weighted MRT sequences of 4 different patient vertebrae. For faster calculation slices were rescaled to a resolution of 128×128 pixels, where 1 pixel corresponds to 1 mm.

2.2 Feature Extraction

We considered a two-dimensional image $\mathbf{I} \in \mathbb{R}^{N \times M}$, where N is the height and M the width of the image. In the first step, we extracted F feature tensors $\mathcal{F}_1 \dots \mathcal{F}_F$ with $\mathcal{F}_f \in \mathbb{R}^{N \times M \times L_f}$, where L_f is the length of the f-th feature response. Before features were fed into the neural network, they were normalized to accomodate for different scales. In our experiments we extracted Gabor wavelet features on 5 scales with 8 orientations, and used the local gray level as a second type of feature.

2.3 Feature Integration by a Reservoir Network

Each of the 4 network layers is a time-discrete network model with a state vector $\mathbf{x} \in \mathbb{R}^X$, where X is the number of neurons, which was set to 100 in our experiments. The state update equation at time step t is given by

$$\mathbf{x}(t+1) = \mathbf{W} \cdot g(\mathrm{diag}(\mathbf{a})\mathbf{x}(t) + \mathbf{b}) + \mathbf{W}_u \mathbf{u}(t)\,. \tag{1}$$

The matrix $\mathbf{W} \in \mathbb{R}^{X \times X}$ gathers the weights of the connections. The vector $\mathbf{u}(t)$ resembles the external input at time step t and $\mathbf{W}_u$ is the input weight matrix. The activation function g is applied component-wise to the vector $\mathbf{x}(t)$ to calculate the firing rates of the neurons and is given by

$$g(x) = \frac{1}{1 + \exp(-x)}\,. \tag{2}$$

The vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^X$ allow for adjustment of a neuron's firing rate. Before the network was used for image interpretation, weights were initialized by drawing uniformly from [-0.02, 0.02] and $\mathbf{a}$ and $\mathbf{b}$ were optimized by intrinsic plasticity [9], which drives the firing rate distribution of the network towards an exponential distribution with fixed mean μ and maximizes the neurons' output entropy under

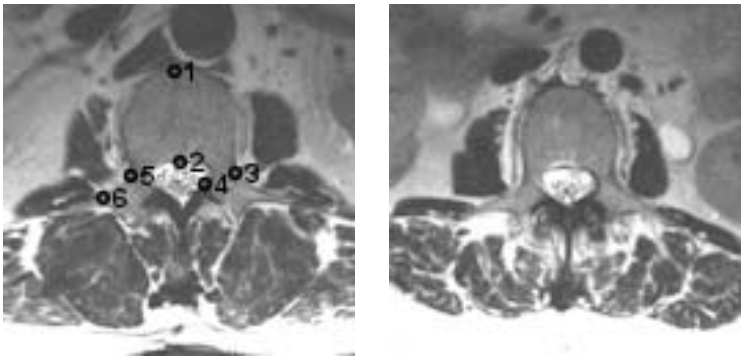


Fig. 1. The labelled image on the left hand side shows the points that were to be detected in the image on the right hand side.

the constraint of the fixed mean. Plasticity was performed by presenting a typical input image once and update **a** and **b** in every time step t according to

$$\Delta b_i(t) = \zeta \cdot \left(1 - \left(2 + \frac{1}{\mu}\right) g(x_i(t)) + \frac{1}{\mu} g(x_i(t))^2\right) , \tag{3}$$

$$\Delta a_i(t) = \zeta \cdot \left(\frac{1}{a} + x_i(t) - \left(2 + \frac{1}{\mu}\right) x_i(t) g(x_i(t)) + \frac{1}{\mu} x_i(t) g(x_i(t))^2\right) , \tag{4}$$

where μ is the target mean and ζ the learning rate. We use $\mu = 0.2$ and $\zeta = 0.001$ in our experiments.

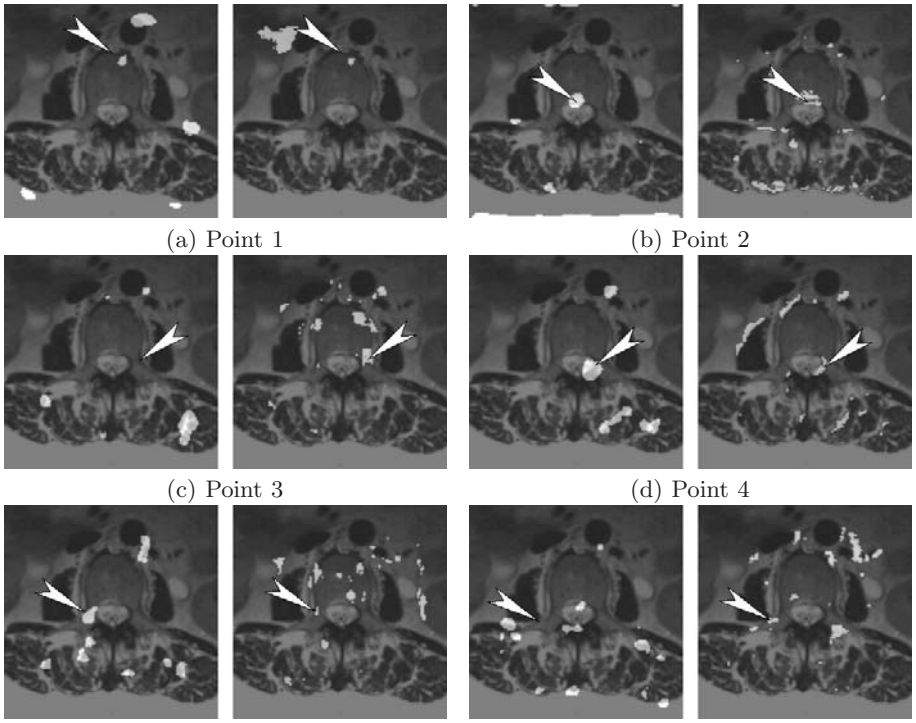
Using update equation (1) we then collected X neuron states for each layer at each point of the image. Every time step t resembles one point of the image if they are sequentially fed into the network. These neuron states defined a combined feature tensor $\mathcal{X} \in \mathbb{R}^{N \times M \times 4X}$.

2.4 Similarity measure and Experimental Setup

To compare two points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ we calculated their similarity S as the reciprocal of the squared error of the respective features:

$$S([\mathcal{X}]_{x_1 y_1}, [\mathcal{X}]_{x_2 y_2}) = \frac{1}{\sum_k \left([\mathcal{X}]_{x_1 y_1 k} - [\mathcal{X}]_{x_2 y_2 k}\right)^2} \tag{5}$$

The task in our experiments was to calculate similarities for 6 distinct landmarks of a labelled vertebra in a target image, by calculating the respective similarities for every point in the target image using either Gabor wavelet features only, or Gabor wavelet features and the local gray level combined by the neural network. Fig. 1 shows the points that were to be detected and the test image.



(a) Point 1 (b) Point 2

(c) Point 3 (d) Point 4

(e) Point 5 (f) Point 6

Fig. 2. Similarity images for the six points (marked by arrows) of the labelled image. Results for Gabor wavelets are on the left hand side, results for the neural network integration are on the right hand side.

3 Results

We display the results by overlaying similarity images and input images to show where high similarities are present (see Fig. 2). For better viewability we applied a threshold to the similarity images to only show the location of local similarity maxima. The threshold was selected in a way that 99% of the gray values in the similarity image were below the threshold. For both experiments local similarity maxima were at least close to the correct location for points 1, 2, 4, and 5. The Gabor wavelets did not produce a maximum near the correct locations for points 3 and 6, while the network processed features were able to detect those points.

4 Discussion

The results show that Gabor wavelet similarity pictures usually produce broad maxima, which correspond to the Gaussian nature of the Gabor wavelets, while the inclusion of gray level as a second feature and the processing by the recurrent

network results in locally bounded maxima. Our experiments indicate that combination of several feature types using a neural network might enable a bunch graph matching algorithm to work on MR data as well as it does on CT data. The improvement over Gabor wavelets alone is based on the combination process by the reservoir, which basically does two things: First, it transforms the original features into random higher-dimensional, and potentially higher-order features, and second, it integrates context into the feature vector of a single point, i. e. features of neighboring points are considered as well. The size and influence of the considered context is implicit within the random weights of the network.

A problem with this approach, however, is the large amount of calculation time required by the neural network as complexity grows quadratically with network size X. This is bothersome, because much larger reservoirs, which are commonly used in reservoir computing, usually produce even better results.

Even though feature quality is hard to judge based on our limited experiments, the results are promising and motivate further research in this direction. In the next step we will expand our evaluation to more data sets as well as to different MR sequences. Performance might further be improved by using even more than two feature types. Additionally, supervised training of the reservoir might overcome the problems with the small network size.

References

1. Liu J, Gao W, Huang S, Nowinski WL. A Model-Based Semi-Global Segmentation Approach for Automatic 3-D Point Landmark Localization in Neuroimages. IEEE Transactions on Medical Imaging. 2008;27(8):1034–1044.
2. Wörz S, Rohr K. Localitaion of Anatomical Point Landmarks in 3D Medical Images by Fitting 3D Parametric Intensity Models. Medical Image Analysis. 2004;10(1):41–58.
3. Roeschies B, Winter S. Detection of Vertebrae in CT slices by Bunch Graph Matching. In: Proceedings of the European Congress for Medical and Biomedical Engineering. vol. 22; 2008. p. 2575–2578.
4. Wiskott L, Fellous JM, Kruger N, von der Malsburg C. Face Recognition by Elastic Bunch Graph Matching. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1997;19(7):775–779.
5. Gabor D. Theory of Communication. J IEE. 1946;93(26):429–457.
6. Graves A, Fernández S, Schmidhuber J. Multi-Dimensional Recurrent Neural Networks. CoRR. 2007;abs/0705.2011.
7. Jaeger H. The "echo state" approach to analysing and training recurrent neural networks. German National Research Center for Information Technology; 2001. 148.
8. Natschläger T, Maass W, Markram H. The Liquid Computer - A Novel Strategy for Real-Time Computing on Time Series. Special Issue on Foundations of Information Processing of TELEMATIK. 2002;8(1):39–43.
9. Triesch J. Synergies between Intrinsic and Synaptic Plasticity in Individual Model Neurons. In: Saul LK, Weiss Y, Bottou L, editors. Advances in Neural Information Processing Systems 17. MIT Press; 2005. p. 1417–1424.