

Diagonalized Nearest Neighbor Pattern Matching for Brain Tumor Segmentation

David T. Gering¹

¹ GE Medical Systems, Applied Science Lab
P.O. Box 414, W875, Milwaukee, WI 53201
david.gering@med.ge.com

Abstract. A new method is proposed for automatic recognition of brain tumors from MRI. The prevailing convention in the literature has been for humans to perform the recognition component of tumor segmentation, while computers automatically compute boundary delineation. This concept manifests as clinical tools where the user is required to select seed points or draw initial contours. The goal of this paper is to experiment with automating the recognition component of the image segmentation process. The main idea is to compute a map of the probability of pathology, and then segment this map instead of the original input intensity image. Alternatively, the map could be used as a feature channel in an existing tumor segmentation method. We compute our map by performing nearest neighbor pattern matching modified with our novel method of “diagonalization”. Results are presented for a publicly available data set of brain tumors.

1 Introduction

[2] introduced the concept of segmenting brain tumors not by focusing on features of tumors, but rather by recognizing deviations from normalcy. Under this philosophy, all training is performed on healthy tissue rather than pathology. Such an approach seeks answers to two guiding questions:

1. What is normal?
2. How is abnormality measured?

One possible solution is to process a training set of example instances of normal images. Given a univariate, normally-distributed, random process, the answers to our two guiding questions are straightforward: normalcy is defined as the population mean, and abnormality is measured as some distance from the mean. The units of measurement for this distance should be standard deviations because a Gaussian process is fully characterized by its mean and standard deviation. For variable x with mean μ and standard deviation σ , expressing distance in this way is commonly known as the Mahalanobis distance:

$$d_1 = \sqrt{\frac{(x - \mu)^2}{\sigma^2}} \quad (1)$$

Next, consider a multivariate process of n correlated variables. Combining the variances and covariances into a covariance matrix Σ , we have:

$$d_n = \sqrt{(\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu})} \quad (2)$$

In the absence of an extensive training population, a definition for normal can be derived from an exploitation of symmetry. For example, it has been proposed that computer-aided diagnosis algorithms for detecting breast and respiratory lesions could exploit left/right symmetry to define normal as the healthy breast or lung. (See [4, 5] for survey articles.) In practice, however, texture from a single healthy breast has been insufficient to capture all the variability, requiring a training set of many scans. We perform experiments here to judge how well normal brain anatomy can be defined as the healthy hemisphere. The problem of recognizing brain tumors may be better suited for exploiting symmetry because the application is for treatment planning rather than screening. Consequently, while breast tumors can appear minutely small on a routine screen, brain tumors tend to not be scanned until their size has grown sufficiently large to become symptomatic.

With symmetry providing examples of normal texture, abnormality can be measured using an appropriate distance metric such as the sum-of-squares distances for a Euclidean space. This leads us naturally to the method of nearest neighbor pattern recognition (NNPM), developed below. The main idea is to compute a map of the probability of pathology, and then segment this map instead of the original input intensity image. Alternatively, the map could be used as a feature channel in an existing tumor segmentation method, such as [6].

2 Method

2.1 NNPM Algorithm

A simple pattern matcher can be constructed from two elements: a container and a comparator. The container holds a set of template patterns, and the comparator computes a distance value, according to an appropriate metric, between each template and the sample under study. The template corresponding with the smallest distance is the nearest neighbor to the sample. Classification can be accomplished with NNPM by classifying the sample by assigning it the label associated with its nearest neighbor [1]. We will adapt NNPM for use as a means of measuring deviations from normalcy.

For our application, define a sample to be a small rectangular window surrounding a certain voxel of the patient's image. Let there be a different container C_i of tem-

plates T_j for each sample S_i in the patient image. Then perform the following algorithm:

```

For each sample  $S_i$  in the patient image:
  For each template  $T_j$  in container  $C_i$ :
    Compute disparity between  $S_i$  and  $T_j$ 
  Record the lowest distance as pixel  $i$  of the result

```

We next consider how NNPM can be used to answer our two guiding questions of what is normal, and how to measure abnormality.

2.2 Measuring Abnormality with NNPM

Let us express the above algorithm mathematically. The method searches for the template with the smallest distance:

$$d_i = \min_{j \in C_i} d_{ij} \quad (3)$$

We next need to define d_{ij} : the distance between the i^{TH} sample in the image, and the j^{TH} template in C_i . We adapt equation 2 by treating each variable within a window as independent, and defining “normal” as the reference value instead of the mean. Instead of normalizing with standard deviations, we normalize with window size W to accommodate comparing the results achieved using various window sizes. These substitutions result in the following equation, which is essentially the root-mean-squared error. Let $S_i[k]$ represent the k^{TH} voxel of the i^{TH} sample, and let $T_j[k]$ represent the corresponding voxel in the j^{TH} template.

$$d_{ij} = \sqrt{\frac{\sum_{k=1..W} (S_i[k] - T_j[k])^2}{W}} \quad (4)$$

Combining the above two equations produce a mathematical expression of the algorithm, given our metric for measuring abnormality:

$$d_i = \min_{j \in C_i} \sqrt{\frac{\sum_{k=1..W} (S_i[k] - T_j[k])^2}{W}} \quad (5)$$

2.3 Defining Normal with NNPM

NNPM defines normal as the set of templates in each container C_i . Each template is an example of normal texture that one would expect to find within the window of W pixels surrounding the i^{TH} voxel of the patient’s image. Since no probability distributions are fit to these templates, building collections of them is straightforward. However, enough templates must be gathered into each container to sufficiently span

the space of normal variation within a window, and none must be examples of abnormal texture near voxel i .

For example, if all C_i were identically all windows within a reference image of a healthy brain, the algorithm would effectively search an entire reference image for the template window that best fits a given window in the patient image. However, by searching the *entire* reference image, spatial information – the location of voxel i – is ignored. Consequently, in a search for deviations from normalcy, finding a dark window where one would expect a light window should be considered abnormal. But if the reference image contained a dark window anywhere, then the algorithm would consider any dark windows in the patient image to be permissible.

Therefore, a more plausible choice of C_i would be the window surrounding the one voxel of the reference image that exhibits the best correspondence with voxel i of the patient image. Correspondence would need to be established by defining a mapping from voxels in the patient image to voxels in the reference image. Such a mapping could be computed as a linear or affine transform using rigid registration, or as a polynomial function or vector displacement field using non-rigid registration. Either way, robustness to registration errors could be introduced by expanding C_i to include all windows centered on the small set of neighboring voxels surrounding the one voxel with the best correspondence. The algorithmic time complexity would then be $O(NMW)$, where N is the image size, W is the window size, and M is the neighborhood size, and $M, W < N$.

How well does a single reference image capture the extent of normal variation within a population? With this thought in mind, perhaps a better approach to defining C_i would involve not one reference image, but a set of images that have been selected to be representative of the complete population. Call this the *training set* of images, and define C_i to include all templates defined as windows surrounding the set of voxels within the neighborhood surrounding the one voxel in each image of the training set that exhibits the best correspondence with voxel i of the patient's image. The time complexity of this algorithm scales linearly with the training set size: $O(NMWT)$. While a larger atlas alleviates the need for a larger search neighborhood, no search neighborhood is as good as a more complete atlas, especially for expressing concepts such as the vessels which rarely appear in exactly the same place on any two scans, but always occur in the same general area.

2.4 Defining Window Size for NNPM

Consider selection of the window size W . For the foregoing discussion, define *micro-texture* to refer to the normal intensity patterns found over small regions, and *macro-texture* to refer to the patterns spread over large areas. The optimal choice of window size is quite application-dependent, as it varies with the interplay between micro- and macro-textures. Selecting a small window size would be adequate to incorporate the context necessary to recognize normal micro-texture, and run times would also be favorable. Large windows, on the other hand, would have the advantage of capturing macro-texture, but they would situate the micro-texture within the macro-texture. That is, if a certain micro-texture pattern could normally be found

anywhere, then enough macro templates would be required to express this fact by exhibiting the certain micro-texture in various situations. Thus, the run-time of the algorithm that correctly uses large window sizes would be dramatically lengthened for two reasons:

1. More time is required to process larger windows
2. More template windows are required to encode more situations

We will refer to this as the double trouble with large window sizes. One way to handle this dilemma would be to isolate the searches for micro- and macro-texture. This will be our goal in the next sections, as we derive our novel *diagonalized NNPM*.

2.5 Multi-scale NNPM

As we seek a means to somehow isolate the searches for micro- and macro-patterns, we acknowledge that there has been much experience within the computer vision community with multi-scale algorithms. When the input data set is downsampled to halve the size of each dimension, 3-D computation with the same window size proceeds 8 times more quickly, and incorporates context from a region 8 times larger. More importantly, at progressively smaller image dimensions, micro-textures become blurred out, allowing the computation to concentrate on macro-textures alone.

In order to avoid the artificial introduction of spurious features, smoothing with a Gaussian kernel (1-4-6-4-1 in our case) must be performed prior to downsampling. This is the purpose of scale-space theory, and in particular, the scaling theorem. Multi-scale analysis for extracting features from a continuum of scales was initiated by [9,8]. The scaling theorem arose when [11] plotted the zero crossings of a Gaussian-smoothed signal over a continuum of scales. The resulting contours form either lines or bowls as the scale progressed from small to large. Thus, the transformation from a fine scale to a course scale can be regarded as a simplification. Fine-scale features disappear monotonically with increasing scale such that no new artificial structures are created at courser scales. Otherwise, it would be impossible to determine if course-scale features corresponded to important fine-scale features, or artifacts of the transformation. In what is known as the scaling theorem, [7] and [12] each proved that the Gaussian kernel uniquely holds this remarkable property.

2.6 Diagonalized NNPM

Completing our derivation of multi-scale NNPM requires some means of combining the results found using fine and course scales. The output of NNPM is a spatial map of distances from normalcy. We create a probability of pathology by normalizing this map to scale from 0 to 1. Let us define the following:

- P(A) = probability of pathology at the highest resolution
- P(B) = probability of pathology at intermediate resolution
- P(C) = probability of pathology at the lowest resolution
- P(A,B,C) = probability of pathology

Operating on the assumption that using multiple scales is successful in isolating micro- and macro-texture, we treat the probabilities of pathology at each resolution as if they were independent. (Although not true in practice, we make this assumption for tractability.) Thus, we can combine the results obtained at each resolution by scaling each result to become a probability map, and then multiplying all the maps:

$$P(A, B, C) = P(A)P(B)P(C) \quad (6)$$

Finally, we must determine the value of the window size parameter, W . Imagine a matrix with a vertical axis of image resolution, and a horizontal axis of window width ($2*r+1$). Figure 1 arranges the resultant images from running NNPM into such a matrix. Instead of using identical window sizes at all scales (such as the red oval in Figure 1 indicates for a window radius of 2), we will prove that the diagonal blue oval is a better choice for us. We label this algorithm, where the window size increases monotonically with decreasing resolution, *diagonalized NNPM*.

Statement:

In the Diagonalized NNPM algorithm, window size increases monotonically with decreasing resolution, resulting in large windows at coarse resolutions.

Reasoning:

- Diagonalized NNPM combines the results obtained at each resolution by scaling each result to become a probability map, and then multiplying all the maps (equation 6).
- The validity of this operation depends on the independence of each map.
- The independence of each map depends on the separation between micro- and macro-texture.
- Micro-texture is most isolated with a small window so that the Gaussian smoothing obscures the micro-features.
- Macro-texture is most isolated with a large window so that a given micro-feature within the window cannot exert a significant influence in the calculation of abnormality (equation 5).
- Thus, multiplicative combination of the maps is best achieved with window sizes that increase with courser resolutions.

QED

3 Results

We performed experiments by running diagonalized NNPM on every case in the tumorbase [7]. The depicted results were generated by defining normal as the two best corresponding slices (computed by symmetry across the midline) from the healthy hemisphere of the same patient. The diagonalization is performed using the following set of window radii from fine to course resolution: $\{1, 1, 2, 2\}$. The segmentation is performed fully automatically by applying a threshold just above the noise floor (the 1% level), and keeping the largest island in the intracranial cavity.

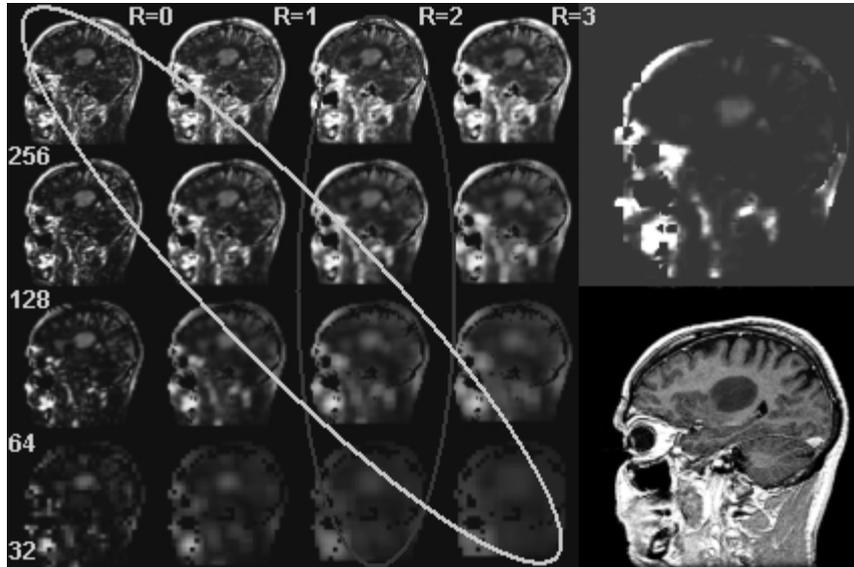


Fig. 1. The diagonalization matrix is shown on the left, the single abnormality map computed from diagonalization is shown on the upper right, and the resulting segmentation of this abnormality map is overlaid in red on the lower right. The red oval represents basic multi-scale NNPM for a window size with radius 2, while the blue oval depicts diagonalized NNPM. (Color version available at <http://www.ai.mit.edu/people/gering/>)

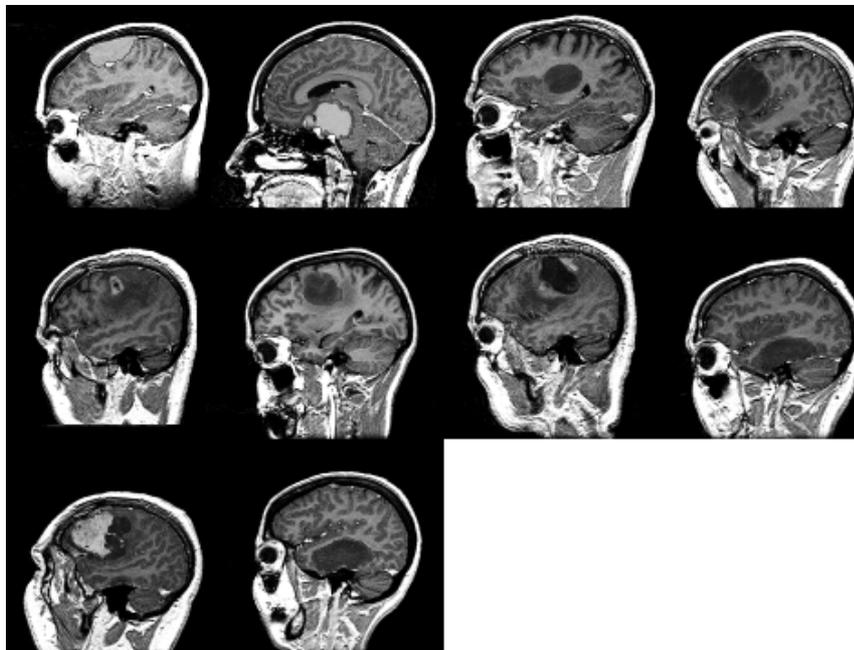


Fig. 2. From left to right are shown the red results overlaid on cases 1-10 of the tumorbase.

4 Discussion

In most of the cases, the boundary delineations of fully automatic segmentation using diagonalized NNPM are too inaccurate for clinical usage. Regardless, the results are encouraging given the goal of this paper, which is to solve the *recognition* problem for brain tumors. Existing methods have largely focused on boundary delineation, leaving the recognition task for humans. With the exception of only one (case #7) of the 10 cases in the tumorbase, diagonalized NNPM correctly recognized the tumor well enough to initiate the boundary delineation process using one of the existing methods. Together, diagonalized NNPM and these methods can form an end-to-end solution for automatic recognition and delineation of brain tumors. For more detailed derivations and results, we refer the reader to [3].

5 Acknowledgements

We sincerely thank Tomas Lozano-Perez for initial discussions proposing this work.

References

1. R.O. Duda, P.E. Hart, D.G. Stork. *Pattern Classification*. John Wiley & Sons, 2001.
2. D.T. Gering, W.E.L. Grimson, R. Kikinis. "Recognizing Deviations from Normalcy for Brain Tumor Segmentation". In: T. Dohi, R. Kikinis, eds. *Medical Image Computing and Computer-Assisted Intervention*. Tokyo, Japan: Springer, 2002; 388-395.
3. D.T. Gering. "Recognizing Deviations from Normalcy for Brain Tumor Segmentation". Ph.D. Thesis, Massachusetts Institute of Technology, 2003 (<http://www.ai.mit.edu/people/gering/>).
4. M.L. Giger. "Computer-aided Diagnosis of Breast Lesions in Medical Images". *Computing in Science & Engineering* Sept-Oct 2000; 2:39-45.
5. B.V. Ginneken, B.M.H. Romeny, M.A. Viergever. "Computer-Aided Diagnosis in Chest Radiography: A Survey". *IEEE Trans Med Imaging* December 2001; 20:1228-1241.
6. M.R. Kaus, S.K. Warfield, A. Nabavi, P.M. Black, F.A. Jolesz, R. Kikinis. "Automated Segmentation of MR Images of Brain Tumors". *Radiology* 2001; 218:586-591.
7. J. Koenderink. "The Structure of Images". *Biological Cybernetics* 1984; 50:363-370.
8. D. Marr, E. Hildreth. "Theory of Edge Detection". *Proceedings of the Royal Society of London* 1980; B207:187-217.
9. A. Rosenfeld, M. Thurston. "Edge and Curve Detection for Visual Scene Analysis". *IEEE Transactions on Computers* 1971; 20:562-569.
10. Surgical Planning Lab. <http://splweb.bwh.harvard.edu:8000>.
11. A. Witkin. "Scale Space Filtering". In: *Proc. International Joint Conference on Artificial Intelligence*. Karlsruhe: 1983; .
12. A.L. Yuille, T.A. Poggio. "Scaling Theorems for Zero Crossings". *IEEE Transactions on Pattern Analysis and Machine Intelligence* January 1986; 8:15-25.