Skin Lesion Segmentation Using Clustering Techniques

**M. Emre Celebi, Wenzhao Guo and
Y. Alp Aslandogan**
Department of Computer Science and Engineering,
University of Texas at Arlington
Arlington, TX 76019-0015 USA
{celebi,alp}@cse.uta.edu

Paul R. Bergstresser
Department of Dermatology,
University of Texas Southwestern Medical Center at Dallas
Dallas, TX 75390-9003 USA
Paul.Bergstresser@UTSouthwestern.edu

Abstract

Cluster analysis has been widely used in various disciplines such as pattern recognition, computer vision, and data mining. In this work we investigate the applicability of two spatial clustering algorithms, namely DBSCAN and STING, to a new problem domain: Color segmentation of skin lesion (tumor) images. Automated segmentation is a key step in the computerized analysis of skin lesion images since the accuracy of the subsequent steps (feature extraction, classification, etc.) crucially depends on the accuracy of this very first step. In this paper, we develop two unsupervised methods for segmentation of skin lesion images: one based on DBSCAN clustering algorithm and the other based on STING clustering algorithm. Experiments on a database of over hundred skin lesion images show that DBSCAN-based segmentation algorithm performs significantly better than the STING-based one.

Introduction

In this work, we examine the applicability of two spatial clustering methods, namely DBSCAN and STING, to a novel domain: Biomedical image data mining. In particular, we explore whether these clustering algorithms can be adapted to solve one of the essential problems of image data mining, accurate identification of significant regions in an image.

Data preparation is typically the least formalized, the most domain-dependent, and the most time consuming part of the knowledge discovery process (Huan and Motoda 1998; Dasu and Johnson 2003). The first step in the preparation of biomedical image data is typically the identification (segmentation) of significant objects such as tumors or tissue fragments. Identification/segmentation is crucial for the subsequent mining tasks, because accurate extraction of significant features relies on this step.

In this paper, we present two unsupervised methods for segmentation of skin lesion images: one based on DBSCAN clustering algorithm and the other based on

---

STING clustering algorithm. These methods allow separation of the healthy skin from the lesion and identification of the subregions inside the lesion with variegated coloring. The number of color regions identified can be fed into further classification and association mining tasks.

The rest of the paper is organized as follows: We first give an introduction to the DBSCAN algorithm. We then discuss our adaptation of this method to the particular domain of skin lesion images. We describe the preprocessing steps before the image is submitted to the clustering module. We then explain the post-processing step which produces a region partitioning that is congruent with human perception. Next we describe our experimental evaluation methodology and the experimental results. Then we describe the application of the STING algorithm in the same domain and present the results. After a discussion of related work, we conclude the paper with a summary of lessons learned and future research.

DBSCAN Algorithm

In this section, we briefly introduce the DBSCAN algorithm. DBSCAN (Density Based Spatial Clustering of Applications with Noise) is a density-based clustering algorithm that is designed to discover clusters and noise in a spatial data set (Ester et al. 1996). It has two major parameters: *Eps* and *MinPts*. The neighborhood of an object within a radius *Eps* is called the *Eps-neighborhood* of the object. If the *Eps-neighborhood* of an object contains at least *MinPts* objects, then this object is called a core object. To find a cluster, DBSCAN starts with an arbitrary object *o* in the data set. If the object *o* is a core object w.r.t. *Eps* and *MinPts*, a new cluster with *o* as the core object is created. The algorithm continues growing the clusters by adding to the clusters all objects that are density-reachable from the core object.

GDBSCAN (Generalized DBSCAN) algorithm, generalizes the two parameters of the DBSCAN algorithm so that it can cluster point objects as well as spatially

extended objects according to both their spatial and non-spatial attributes (Sander et al 1998). The neighborhood of an object is defined by a binary predicate *NPred*. The *NPred-neighborhood* of an object is defined as the set of objects for which the predicate *NPred* is true. A second predicate *MinWeight* of a set of objects *S*, is defined such that it is true iff the weighted cardinality of the set *S*, wCard(S), is greater than or equal to the minimum cardinality, *MinCard*.

Skin Lesion Segmentation using DBSCAN

Here, we follow a region-based segmentation approach. The goal is to partition an image into disjoint regions that correspond to healthy skin and subregions inside lesion. Figure 1 shows the flowchart of the segmentation procedure. There are two major steps involved. First, the image is split into smaller regions until all the regions meet the homogeneity criteria determined by the threshold for splitting. Then, these regions are merged to form the final regions by the DBSCAN algorithm.

The segmentation algorithm employs an iterative, coarse-to-fine strategy. The aim of the first iteration is to identify the lesion border. During the following iterations, the algorithm moves towards the inside of the lesion and identifies subregions with variegated coloring. This approach allows the algorithm to dynamically adjust the parameters during various stages of the procedure.

The following subsections will describe the steps involved in the segmentation procedure.

Preprocessing

Skin lesion images often contain extraneous artifacts such as skin texture and hair that make the segmentation more difficult. In order to reduce the effects of these artifacts, we have smoothed the images with a 5 x 5 median filter (Gonzalez and Woods 2002). This is filter has the advantage of preserving the edges well enough for accurate border detection (Schmid 1999).

Region Splitting

Region splitting is a top-down process. A region is split into four subregions if it fails to satisfy the homogeneity criterion. Splitting process starts from the highest-level region which is the whole image to be segmented. Each region in the image is represented by its mean color calculated in the RGB color space. To determine the homogeneity of a region, the Euclidean distances between the region and each of its four potential subregions are computed. If any of the color distances computed above is greater than the threshold for splitting, the region will be split. Otherwise, the region remains unchanged. The

splitting process goes on recursively until each region is either found to be homogeneous or too small to be split further.
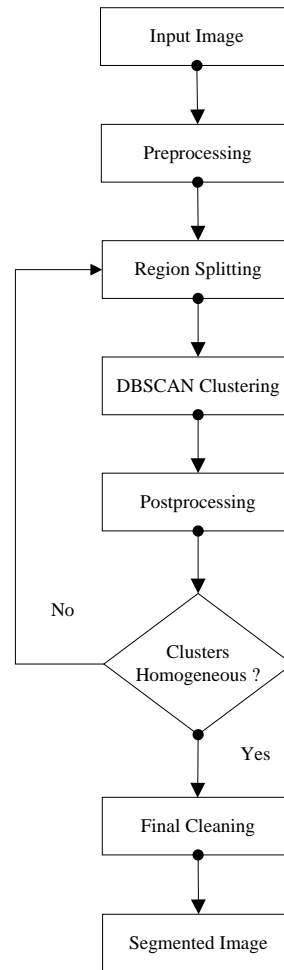

Figure 1. Flowchart of the segmentation procedure

The threshold for splitting is a very important parameter. If it is set too high the accuracy of the segmentation will decrease. The result will show rectangular borders around the segments. On the other hand, if it is set too low, the image might be over-segmented with most of the segments being as small as a few pixels. The program automatically determines the threshold for splitting as follows.

First, we determine the color of the healthy skin. This is done by taking four windows of size 10 x 10 pixels from four corners of the image and calculating the median color of the pixels. We use this median color as the color of the healthy skin. Note that median color is used instead of mean to reduce the interference of hairs (Xu et al. 1999). Then, we divide the image into 100 subregions and use each of the subregions as a sample of the image. Finally, we calculate the color distances between the healthy skin color and each of these samples and sort these values. The threshold for splitting is determined from the significant gaps among these values. In the following discussion, the

distance between the healthy skin color and a sample n will be simply referred to as the value of sample n.

Let us demonstrate the threshold selection using Figure 2. Here, there is a significant gap between the values of sample n and sample $n+1$. If the amount of this gap is greater than a threshold T_G and the value of sample n is also greater than a threshold T_C, we let the threshold for splitting to be the average of values of sample n and sample $n+1$. The thresholds T_G and T_C are determined empirically from a set of representative images as described in the Experimental Results section.
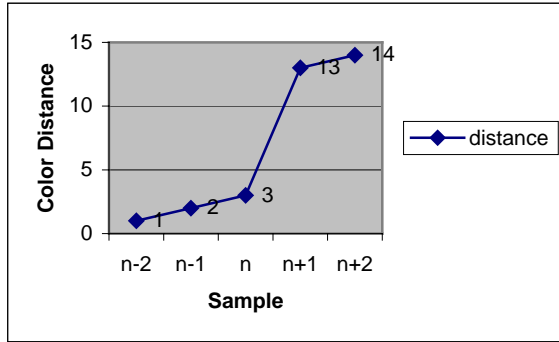

Figure 2. Demonstration of threshold selection

Once the splitting process is completed, a graph is constructed to represent the regions and their connectivity. Two regions are considered connected if they share a common boundary. The graph is represented by an adjacency matrix.

DBSCAN Clustering

While GDBSCAN generalizes DBSCAN to a higher level in order to be able to deal with a wider range of problems, we will carry out a specialization of GDBSCAN so that it fits our purposes. In our case, the clustering algorithm is applied to a set of regions generated as a result of splitting process and our objective is to group these regions to form larger regions that have certain amount of color homogeneity.

We define the two parameters of GDBSCAN algorithm, *NPred* and *MinWeight,* as follows. Since each region has a non-spatial attribute, color mean, and a spatial attribute, location in the image, the binary predicate *NPred* is defined as "color distance between two regions is smaller than *Eps* **AND** the two regions are connected" where *Eps* is a threshold of color distance determined from a set of representative images. For instance, given a region *R1*, a region *R2* is in the *NPred-neighborhood* of *R1* iff the color distance between *R1* and *R2* is smaller than *Eps* **AND** *R1* and *R2* are connected. The function wCard(**.**) is defined to return the total number of pixels in the *NPred-neighborhood*. *MinCard* is defined as a threshold of number of pixels, which is obtained experimentally. The predicate *MinWeight* is therefore defined as "total number of pixels in the *NPred-neighborhood*, given by wCard(**.**), is greater than *MinCard* pixels".

The DBSCAN algorithm starts from an arbitrary region *R*. First, using the *Npred* predicate, all connected regions that have a color distance less than *Eps* to *R* are determined. These regions form the *NPred-neighborhood* of *R*. The total number of pixels in the *NPred-neighborhood* is calculated using wCard(**.**). If the total number of pixels is greater than *MinCard*, a new cluster with region *R* as the core object is created. Then, all directly density-reachable regions that are reachable from the regions in the *NPred-neighborhood* are added to the current cluster. The algorithm continues iteratively until no new region satisfies the predicates.

A sample lesion and the result of the initial DBSCAN clustering on this lesion are shown in Figure 3. The black regions in Figure 3b) represent the noise. They cannot be integrated into any cluster because they fail the predicate test. Notice that most of the noise is along the border of the lesion. This can be explained as follows. There is a much higher color variation in the thin region along the border when compared to the other regions. Because of this, during the region splitting, this region is divided into much smaller subregions. Since they have a high color distance to the neighboring regions, these tiny subregions are unlikely to pass the predicate test to form a new cluster or to join an existing one.
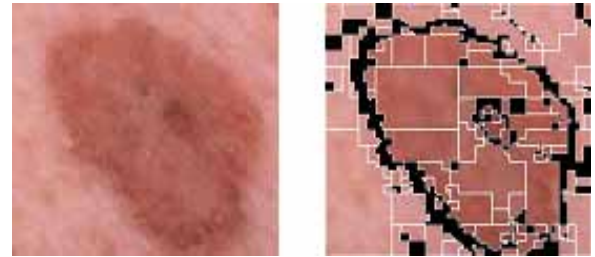

Figure 3. a) Sample image   **b) Before postprocessing**

DBSCAN is designed to discover noise as well as clusters in a spatial data set. Also, as can be seen in Figure 3b), the number of clusters obtained from the first iteration is much more than the number of color regions that can be identified by humans. Therefore, some postprocessing is necessary.

First, the noise pixels are merged with neighboring clusters that are closest in color. Then, clusters that are closer (in color) to each other than a threshold are merged. The value of this threshold is determined by multiplying the threshold for splitting by a factor. This factor is proportional to the amount of color variation present in the initial regions. The amount of color variation in a region can be represented by the color mean absolute deviation (MAD). A region with less color variation has a smaller color MAD. In such a region, the threshold value is smaller and the merging criterion is more strict which allows the algorithm to detect subtle color variation.

More Iterations of DBSCAN

As a result of the first iteration of DBSCAN, we have identified the lesion border. Figure 4(a) shows the image in Figure 3(a) after the first iteration. Our next goal is to go inside the lesion and identify subregions with color variegation if they exist.

If a cluster satisfies any of the following three conditions, more iterations of DBSCAN is performed: (i) the size of the cluster is larger than $1/10^{th}$ of the original image (ii) the color distance between the cluster and the healthy skin is larger than a threshold T_H (iii) the amount of color variation in the cluster is larger than a threshold T_V. The thresholds T_H and T_V are determined empirically from a set of representative images as described in the results section.

In order to avoid over-segmentation, more than two extra iterations of DBSCAN is not allowed.

Final Cleaning

Despite the postprocessing performed after each iteration of DBSCAN, the resulting images may still have more regions than humans can perceive. Therefore, one final cleaning step is necessary. During this step, if a region is smaller than $1/100^{th}$ of the original image it is merged with the nearest cluster. Figure 4(b) shows the final segmentation result of the image in Figure 3(a).

Figure 4. a) After the 1ˢᵗ iter.   b) Final segmentation result

DBSCAN Segmentation Results

In order to fine-tune the algorithm parameters and test the effectiveness of the proposed segmentation method, experiments are performed on a set of 135 clinical images with dimensions subsampled to 256 x 256.

First, to fine-tune the algorithm parameters, 18 representative images are selected from the above set. After fine-tuning the parameters, the algorithm is applied to the remaining 117 images. An expert dermatologist (4ᵗʰ author) visually examined the segmentation results. In 80% of the images the lesion borders were detected successfully. Figure 5 shows a sample of the segmentation results.

To further evaluate the effectiveness of the segmentation, an experiment is performed to compare the number of color regions identified by the algorithm and by the human subjects. Five subjects participated in the experiment. They were graduate students, ranging in age from 22 to 25. No subject had seen the segmentation results before he/she performed the experiment. All subjects had normal or corrected-to-normal vision.
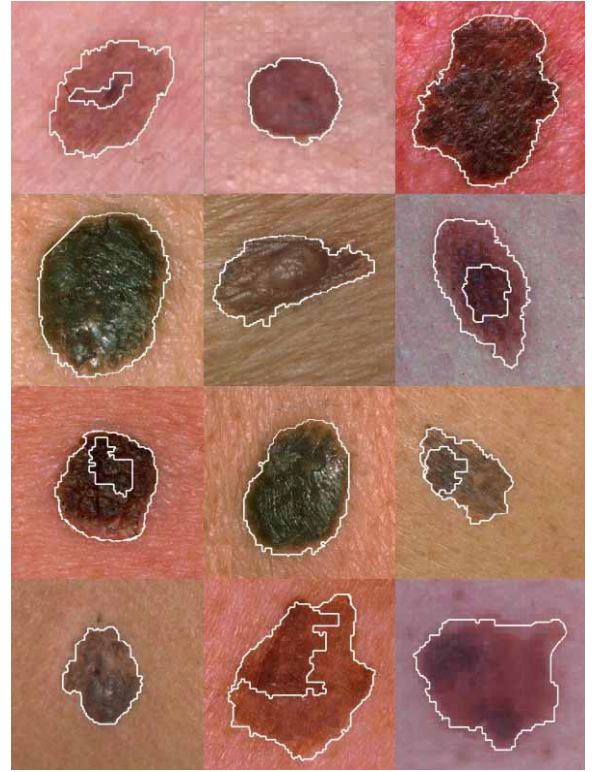

Figure 5. A sample of DBSCAN segmentation results

The number of colors/subregions inside the lesion is used as the evaluation criterion. The kappa statistic of Cohen (Cohen 1960) is chosen as the measure of agreement between the segmentation results and the aggregate human observation obtained by majority vote.

The kappa coefficient equals +1 when there is a complete agreement among the observers. When the observed agreement exceeds chance agreement, kappa is positive, with its magnitude reflecting the strength of agreement. Negative values indicate that the observed agreement is less than chance agreement.

We convert the evaluation criterion (the number of colors/subregions inside the lesion) to ranks as follows: Rank 1 (1 color), Rank 2 (2 colors), and Rank 3 (3 or more colors). In the experiment, no subject observed more than 3 colors/subregions in a lesion. Furthermore, only 6 out of 117 images in the segmentation results have more than 3 identified colors/subregions. Therefore, we assigned rank 3 to the cases where the lesion has 3 or more colors/subregions.

Table 1. Evaluation results

		Human Observation			
		1	2	3	Total
Segmentation	1	2	3	1	6
Result	2	3	58	5	66
	3	2	28	15	45
	Total	7	89	21	117

Table 1 shows the evaluation results. The kappa coefficient, calculated from this table using SAS® System v8.0, is 0.28 (95% confidence interval), showing a significant agreement between the segmentation results and human perception.
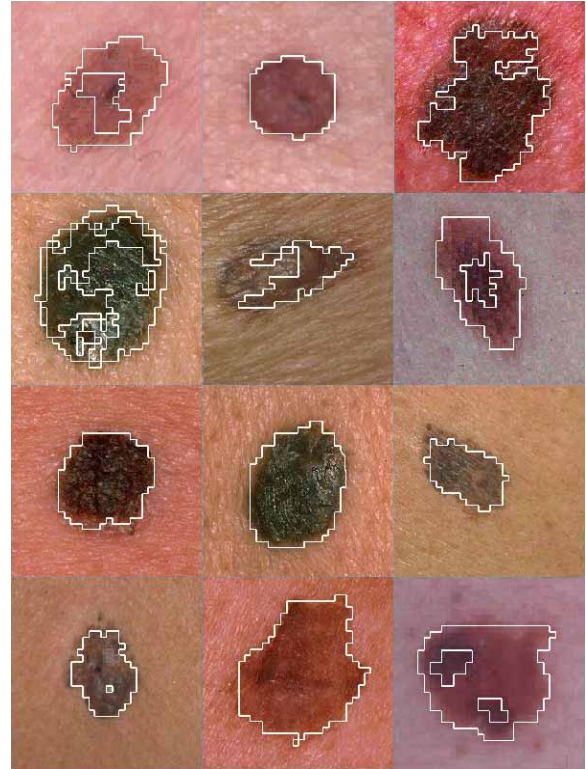
Skin Lesion Segmentation Using STING

STING is a typical grid-based clustering approach that is designed to solve clustering and region-oriented query problems (Wang, Yang and Muntz 1997). It divides the spatial area into rectangular cells and builds a hierarchical structure storing statistical information of cells. Statistical parameters of higher-level cells can be easily computed from the parameters of the lower-level cells. With the hierarchical structure of the grid in hand, a top-down approach can be followed to answer spatial data mining queries. For each cell in the top layer, we compute the likelihood that this cell is relevant to the query at some confidence level to determine the cell's relevancy to the given query. The likelihood can be defined as the proportion of objects that satisfy the query. Processing of the next lower level examines only the remaining relevant cells. The process is repeated until the bottom layer is reached. The final result is formed as the union of qualified bottom level cells.

In order to use STING for skin lesion segmentation, we chose the intensity value of a pixel as the attribute. Histogram analysis is used to provide information for generating region retrieval queries. The rationale is that a significant peak in the histogram corresponds to either a single color region of interest or several regions of interest with the same color. We generate one query for each peak. Answering each of the queries corresponds to discovery of a homogeneous color region. Figure 6 shows a sample of the segmentation results.

The distribution type of pixel intensity is not known before hand and carrying out hypothesis test for each cell in the hierarchy structure of STING is complicated and expensive. Thus, when we compute the confidence interval to determine a cell's relevancy to the query we estimate the proportion range that the pixel intensity falls in a particular range by distribution-free technique Chebyshev inequality.

Once all the cells are marked as either relevant or not relevant, the final result is formed as the union of qualified bottom level cells.

Figure 6. A sample of STING segmentation results

Related Work

Several methods have been developed for segmentation of pigmented skin lesions, most of them being focused on border detection. In this section we review some of the recent approaches.

Xu et al. (Xu et al. 1999) proposed a two-stage approach for unsupervised border detection. First, the color image is reduced to an intensity image which is then thresholded to obtain an approximate border. Then, the border detection is refined using edge information in the neighborhood of the initial border. Zhang et al. (Zhang, Stoecker and Moss 2000) presented an automatic scheme based on two-rounds of radial search technique originally introduced in (Golston et al. 1990). Chung and Sapiro (Chung and Sapiro 2000) proposed a partial-differential equations-based approach using the geodesic active contours model. Metz et al. (Metz et al. 2001) presented a semi-automatic method very similar to Xu et al.'s method that incorporates an interactive selection step to clip the region of interest. She and Fish (She and Fish 2002) proposed a technique based

on edge focusing. A fast snake is used to determine the optimum border by minimizing the energy calculated by the Laplacian of Gaussian (LoG) edge detector. Recently, Rajab et al. (Rajab, Woolfson and Morgan 2004) proposed an approach based on optimal thresholding using an isodata algorithm.

Conclusions and Future Work

We have presented two unsupervised methods for segmentation of skin lesion images based on DBSCAN and STING clustering algorithms. We conclude that while STING algorithm runs faster, DBSCAN algorithm produces more accurate segmentation results.

These methods do not perform well when the image is too hairy. For such images, a preprocessor that can eliminate hairs such as DullRazor™ (Lee et al. 1997) may improve the segmentation results.

Acknowledgments

References

Chung D.H. and Sapiro G. 2000. Segmenting Skin Lesions with Partial-differential-equations-based Image Processing Algorithms. *IEEE Transactions on Medical Imaging* 19(7):763-767.

Cohen J.A. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement* 20(1):37-46.

Dasu T. and Johnson T. 2003. *Exploratory Data Mining and Data Cleaning*. Wiley Publishing Inc.

Ester M., Kriegel H.-P., Sander J., and Xu X. 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, 226-231. Portland, Oregon: AAAI Press.

Golston J.E., Moss R.H., and Stoecker W.V. 1990. Boundary Detection in Skin Tumor Images- An Overall Approach and a Radial Search Algorithm. *Pattern Recognition* 23(11):1235-1247.

Gonzalez R.C. and Woods R.E. 2002. *Digital Image Processing*. Upper Saddle River, New Jersey: Prentice Hall.

Huan L. and Motoda, H. eds. 1998. *Feature Extraction, Construction and Selection: A Data Mining Perspective.* Kluwer Academic Publishers.

Jain A.K., Murty M.N., and Flynn P.J. 1999. Data Clustering: A Review. *ACM Computing Surveys* 31(3):264-323.

Jain A.K. and Dubes R.C. 1988. *Algorithms for Clustering Data*. Upper Saddle River, New Jersey: Prentice Hall.

Lee T. K., Ng V., Gallagher R., Coldman A., and McLean D. 1997. Dullrazor- A Software Approach to Hair Removal from Images. *Computers in Biology and Medicine* 27(6):533-543.

Metz M., Groch W.-D., Curnow J., Ifeachor E.C., and Kersey P.J. 2001. Computer Aided Lesion Border Detection Applied to Macroscopic Images. In *Proceedings of the 4th International Conference on Neural Networks and Expert Systems in Medicine and Healthcare*.

Rajab M.I., Woolfson M.S., Morgan S.P. 2004. Application of Region-based Segmentation and Neural Network Edge Detection to Skin Lesions. *Computerized Medical Imaging and Graphics* 28(1-2):61-68.

Sander J., Ester M., Kriegel H.-P., and Xu X. 1998. Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and its Applications. *Data Mining and Knowledge Discovery* 2(2):169-194.

Schmid P. 1999. Segmentation of Digitized Dermatoscopic Images by Two-Dimensional Color Clustering. *IEEE Transactions on Medical Imaging* 18(2):164-171.

She Z. and Fish P.J. 2002. Boundary Detection of Skin Lesion using a Fast Snake Algorithm. *In Proceedings of the 16th International EURASIP Conference*, 295-297.

Wang W., Yang J., and Muntz R. 1997. STING: A Statistical Information Grid Approach to Spatial Data Mining. In *Proceedings of the International Conference on Very Large Databases (VLDB'97)*, 186-195.

Xu L., Jackowski M., Goshtasby A., Roseman D., Bines S., Yu C., Dhawan A., and Huntley A. 1999. Segmentation of Skin Cancer Images. *Image and Vision Computing* 17(1):65-74.

Zhang Z., Stoecker W.V., and Moss R.H. 2000. Border Detection on Digitized Skin Tumor Images. *IEEE Transactions on Medical Imaging* 19(11):1128-1143.