PROJECTE FINAL DE CARRERA

Eye gazed regions prediction in lung radiography

AUTHOR: Oriol Caudevilla Torras
ADVISOR: Jovan Brankov

2012/13

Escola Tecnica Superior d'Enginyeria de Telecomunicacio de Barcelona

Medical Imaging Research Center at Illinois Institute of Technology

Acknowledgement

I would like to dedicate this lines to thank the precious help and support I have received during the time I have been working on this project. This work could not have been done without being mentored by Professor Jovan Brankov which excellent advises pushed forward my research. Finally, I would like to thank the Centre de Formació Interdisciplinaria Superior at Universitat Politecnica de Catalunya and the Medical Imaging Research Center at the Illinois Institute of Technology for the financial support and the technical resources they have offered me. Their help has been essential for me, I will never forget that they made this project possible.

Resum

Aquest projecte final de carrera es una primera aproximació a la diagnosi assistida de radiografies de pit. Donada una radiografia, es pretén desenvolupar un observador artificial que determini a quines zones de la imatge es fixa un metge al diagnosticar-les. Per a fer-ho, es disposa de una base de dades amb diversos exemples de radiografies i una llista dels píxels que han estat observats pels observadors humans experts, metges. Aquestes dades han estat captades mitjançant tècniques de traçat de mirada quan als metges se'ls demanava que diagnostiquessin les imatges. Aquest treball d'investigació pretén desenvolupar un complet algoritme de aprenentatge estadístic que utilitzi aquestes dades i les processi per a poder predir zones d'interès a les radiografies.

La tècnica de machine learning emprada és una Relevance Vector Machine (RVM), una alternativa al més comú Support Vector Machine (SVM). L'enfoc del problema tractat consisteix en predir el nombre de punts on un observador humà fixa la mirada en cada regió de interès de la imatge. Així mateix, les regions d'interès també son determinades per l'algoritme desenvolupat.

Per dur a terme l'algoritme d'aprenentatge, és necessari realitzar un preprocessament de les imatges per tal de poder extreure característiques (o features) rellevants per al problema que ens ocupa. Aquest preprocessament també ens és útil per a poder segmentar cada imatge en regions d'interès (que com s'explica en aquest document corresponen a zones amb possibles masses). El primer algoritme realitzat és una transformació basada en Smooth Thin Splines (STS) per a alinear les caixes toràciques de cada imatge a un template. D'aquesta forma podem utilitzar la posició de cada regió com una feature per al RVM. A continuació, s'utilitza un procediment de filtrat per a la detecció de possibles cossos anòmals en els pits. Aquest algoritme s'anomena multi-resolution Convergence Index Filtering. Es tracta d'un pas clau per a la detecció de masses clares als pulmons ja que podem obtenir una imatge que ressalta les zones sospitoses. Les imatges filtrades son posteriorment segmentades amb la tècnica watershed. D'aquesta forma s'obtenen les regions de interès que constituiran les mostres d'entrada per al RVM.

Finalment, cada regió de interès constitueix una mostra de la que es mesuren features d'interès per a decidir si dita regió conte algun element interessant per als doctors. Es prenen mesures morfologiques, com el tamany o el perimetre de les regions, i mesures de contrast, com el màxim rang de l'imatge filtrada dins de cada regió. Aquestes son les dades d'entrada per a realitzar un aprenentage estadístic, en particular el RVM que s'ha utilitzat. RVM permet realitzar una estimació del error que hi haurà per a cada mostra, com es veu en aquesta memòria, fet que el fa particularment útil en aquesta aplicació. Cal destacar que s'ha utilitzat l'algoritme de Principal Component Analysis (PCA) per a reduir la dimensionalitat i determinar els features més útils per al aprenentatge.

Els experiments, que es presenten aquí, suggereixen que el Convergence Index Filter és una bona primera aproximació a la detecció de masses en radiografies de pit. A més, el RVM prenent cada regió de interès segmentada amb watershed com a mostra ha resultat ser una bona forma d'enfocar el problema de predir els punts on els metges fixen la seva mirada.

Resumen

Este proyecto final de carrera es una primera aproximación a la diagnosis asistida de radiografías de pecho. Dada una radiografía, se pretende desarrollar un observador artificial que determine en que zonas de la imagen se fija un médico al diagnosticarlas. Para ello se dispone de una base de datos con diversos ejemplos de radiografías y una lista de píxeles que han sido observados por observadores humanos expertos, médicos. Estos datos han sido captados mediante técnicas de trazado de mirada cuando a los médicos se les pedía que diagnosticasen las imágenes. Este trabajo de investigación pretende desarrollar un completo algoritmo de aprendizaje estadístico que utilice estos datos y los procese para poder predecir zonas de interés en radiografías de pecho.

La técnica de machine learning utilizada es una Relevance Vector Machine (RVM), una alternativa al más utilizado Support Vector Machine (SVM). El enfoque del problema tratado consiste en predecir el número de puntos donde un observador humano fija la mirada en cada región de interés de la imagen. Así mismo, las regiones de interés también son determinadas por el algoritmo desarrollado.

Para llevar a cabo el proceso de aprendizaje, es necesario realizar un preprocesado de las imágenes para poder extraer características (o features) relevantes para el problema que nos ocupa. Este preprocesado también nos resulta útil para poder segmentar cada imagen en regiones de interés (que, como se explica en este documento, corresponden a zonas con posibles masas). El primer algoritmo realizado es una transformación basada en Smooth Thin Splines (STS) para alinear las cajas torácicas de cada imagen a un template. De esta forma podemos utilizar la posición de cada región como una feature para el RVM. A continuación, se utiliza un procedimiento de filtrado para la detección de posibles cuerpos anómalos en el pecho. Este algoritmo se llama multi-resolution Convergence Index Filtering. Se trata de un paso clave para la detección de masas claras en los pulmones ya que podemos obtener una imagen que resalta las zonas sospechosas. Las imágenes filtradas son posteriormente segmentadas con la técnica watershed. De esta forma se obtienen regiones de interés que constituirán las muestras de entrada para el RVM.

Finalmente cada región de interés constituye una muestra de la que se miden features de interés para decidir si dicha región contiene algún elemento interesante para los doctores. Se toman medidas morfológicas, como el tamaño o el perímetro de las regiones, y medidas de contraste, como el rango máximo de la imagen filtrada dentro de cada región. Estos son los datos de entrada para realizar un aprendizaje estadístico, en particular el RVM que se ha utilizado. RVM permite realizar una estimación del error que habrá para cada muestra, como se ve en esta memoria, hecho que lo hace particularmente útil en esta aplicación. Cabe destacar que se ha utilizado el algoritmo de Principal Component Analysis (PCA) para reducir la dimensionalidad y determinar los features más útiles para el aprendizaje.

Los experimentos, que se presentan aquí, sugieren que el Convergence Index Filter es una buena primera aproximación a la detección de masas en radiografías de pecho. Además, el RVM tomando cada región de interés segmentada con watershed como una muestra ha resultado una buena forma de enfocar el problema de predecir los puntos donde los médicos fijan su mirada.

Abstract

This final degree project is a first approach to an assisted diagnosis applied to chest x-ray images. Given a x-ray image, we pretend to develop an artificial observer that determines which areas of the image are relevant for a doctor during the diagnosis. We are using a dataset with several examples of chest x-ray images and a list of pixels gazed by expert human observers, doctors. This data is obtained by eye tracking techniques applied while the doctors were asked to diagnose the images. This research has developed a complete statistical learning algorithm that uses the given dataset and uses it for predicting regions of interest in the image.

The machine learning algorithm used is a Relevance Vector Machine (RVM), an alternative to the Support Vector Machine (SVM). The goal we pretend to achieve is determining the number of eye gazes of the doctors in each region of interest of the image. Also, our algorithm will determine what are the regions of interest.

In order to perform the RVM, preprocessing the images is needed to obtain relevant features for the problem we are trying to solve. This previous steps will also be useful for segmenting the image in regions of interest (as it will be seen in this report, each region corresponds to a possible mass in the lungs). First, we use a Smooth Thin Plate (STP) algorithm to align the thoracic cage to a template. Positions of the regions of interest will be used as a features for the RVM. Second, a multi-resolution Convergence Index Filtering is performed for mass detecion in the lungs. This is a key step for bright regions detection because we can obtain an image with the suspicious areas highlighted. Filtered images are then segmented with a technique called watershed, obtaining the regions of interest that will be used as samples for the RVM.

Finally, each region of interest is considered a sample and for each sample some features are taken to decide if it is interesting for doctors or not. Morphological features, such as region size and perimeter, and contrast features, such as the maximum range of the filtered image at each region are the most useful. This data is the input for a statistical learning process, in our case the RVM. RVM allows us to estimate the error at each sample, as it will be shown in this report, which is especially useful for our problem.

Our experiments, exposed here, suggest that the Convergence Index Filtering is a good approach to nodule detection in chest x-ray images. Also, the RVM taking each region of interest segmented with watershed as a sample has turned out to be a useful point of view to the eye gaze prediction problem.

Contents

List of Figures

Chapter 1

Introduction

Artificial interpretation of a x-ray image is one of the most useful AI tools one could imagine. This is one approximation to that last goal. Using eye gaze tracking data, machine learning and image processing techniques we attempt to emulate an expert human observer.

1.1 Introduction to chest x-ray images

1.1.1 Projection radiography

Chest x-ray images are obtained using a technique called projection radiography. X-Rays is an electromagnetic radiation between $3 \cdot 10^{16} Hz$ and $3 \cdot 10^{19} Hz$ with energies that range 100 eV to 100 KeV. That amount of energy is enough to liberate electrons from atoms under x-rays action, which makes them an ionizing radiation.

Projection radiography obtains attenuation maps of the human body using a source and a receptor of x-rays. Radiation cross the human body and reaches the receptor located at the other side of the body. When radiation arrives at the receptor, its energy is lower if it had to cross a bone than if it had to cross fat or air. Dense objects are hard to cross for x-rays. Because of that, radiation energy reception is heterogeneous in the detector. X-ray images are builded with those differences, showing white areas (shadows of radiation) where the rays had to cross a bone and dark areas were radiation was more intense (for example lungs, where there is a lot of air which offers a very little resistance to the radiation).

1.1.2 Initial chest x-ray analysis

General notions of chest x-ray images are necessary to easily make reference to different parts of the human chest. Figure 1.1 shows an example of chest x-ray image where important anatomical parts have been highlighted.

Relevant bones observed in this kind of images conform the thoracic cage: sternum, 12 thoracic vertebrae (center) and 24 ribs. In the image, appear only 22 ribs. Also, the clavicle is is clearly visible in those kind of images. As said in the last section, the black region correspond to de air inside the lungs,down edge of witch shows the diaphragm. Heart is very clearly identifiable as the white centred rounded region. Finally, it is important to identify bronchioles because can be easily confused with a tumour if you do not know where are them. Differencing bronchioles from suspicious masses is one of the problems we will have to face because both appear as rounded bright areas on a chest x-ray images.

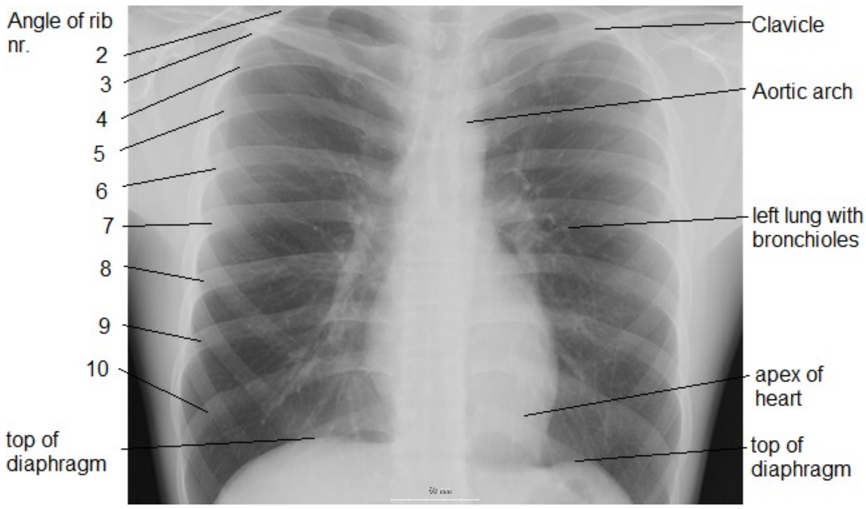

Figure 1.1: Relevant parts of a frontal chest x-ray image

1.2 Examples of visible diseases on a chest x-ray image

Chest x-ray images give a diverse medical information. Doctors are able to detect signs of pneumonia 1.2, tuberculosis 1.3, pleural effusion 1.4, heart failure 1.5, sarcoidosis 1.6, pneumotorax 1.7, lung cancer 1.8, etc. All those diseases show very different symptoms: big white shadows, isolated nodules, abnormal heart size, etc. Some of those signs are very difficult to identify, a subtitle density change could be a tumour.

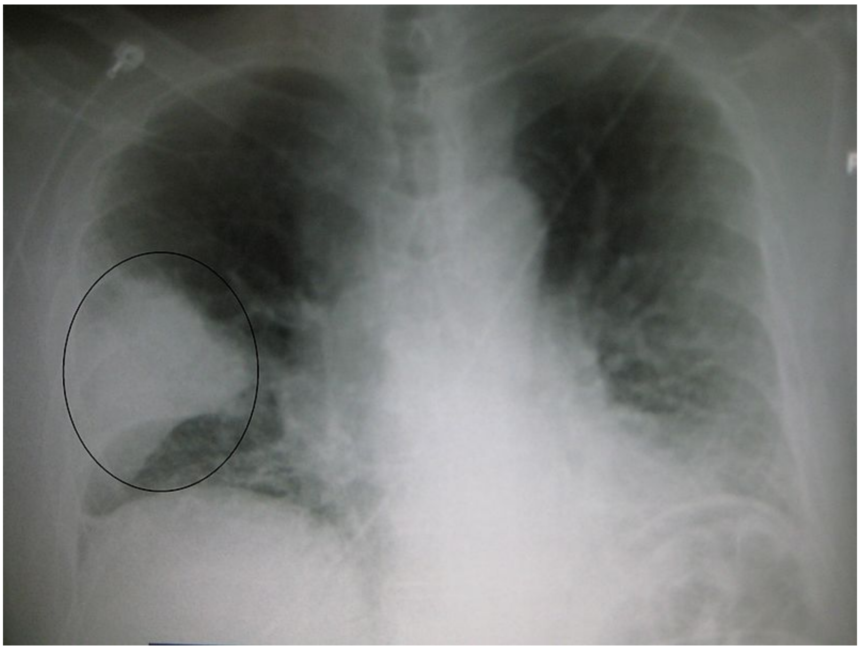

Figure 1.2: A very prominent pneumonia of the middle lobe of the right lung. Author: James Heilman, MD.

1.3 Problem definition and main structure of the algorithm

Now that some basics are introduced, we need to detail what are our objectives and the main structure of the algorithm developed to reach them. Our goal is to predict where a doctor will look to diagnose a chest x-ray image. We are going to use a machine learning algorithm known as Relevance Vector Machine (RVM), an alternative to the more known Support Vector Machine (SVM). The RVM procedure is detailed in the chapter 7.

As a samples for the RVM, we are using regions of interest. This regions of interest correspond to nodules candidates that we can segment after filtering the images with a Convergence Index Filter (CIF). CIF algorithms look for possible masses or anomalous objects in the lungs and are explained in the chapters 3 and 4. Once CIF output is obtained, a watershed segmentation is done to obtain the regions of interest, as said in chapter 5. So, our algorithms will be predicting the number of gazes per region, this way we can know what are the most important areas in the image. Once the regions are obtained and the CIF is performed, some features are extracted for each region. The features extracted are detailed in chapter 6. For all this process, it is necessary to align the images to a template. Chapter 2 explains how the thoracic cages are aligned

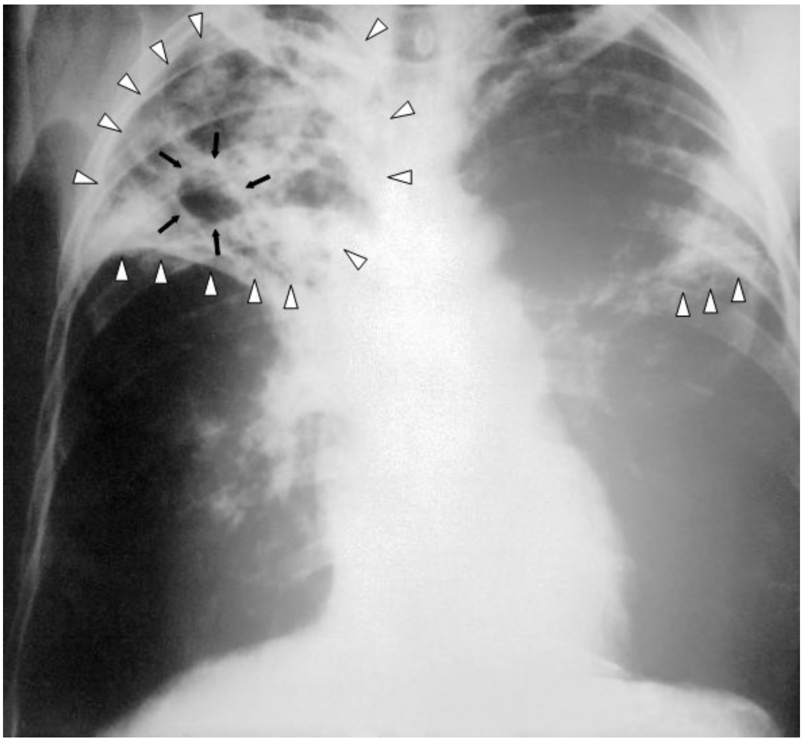

Figure 1.3: Bilateral pulmonary infiltrate (white triangles), and "caving formation" (black arrows) present in the right apical region.The diagnosis is far-advanced tuberculosis. Author: Center for Disease Control and Prevention.

to be compared.

Finally, chapter 8 presents the results of the experiments done with this algorithm, extracts the conclusions and present some future work that should be done to continue the research.

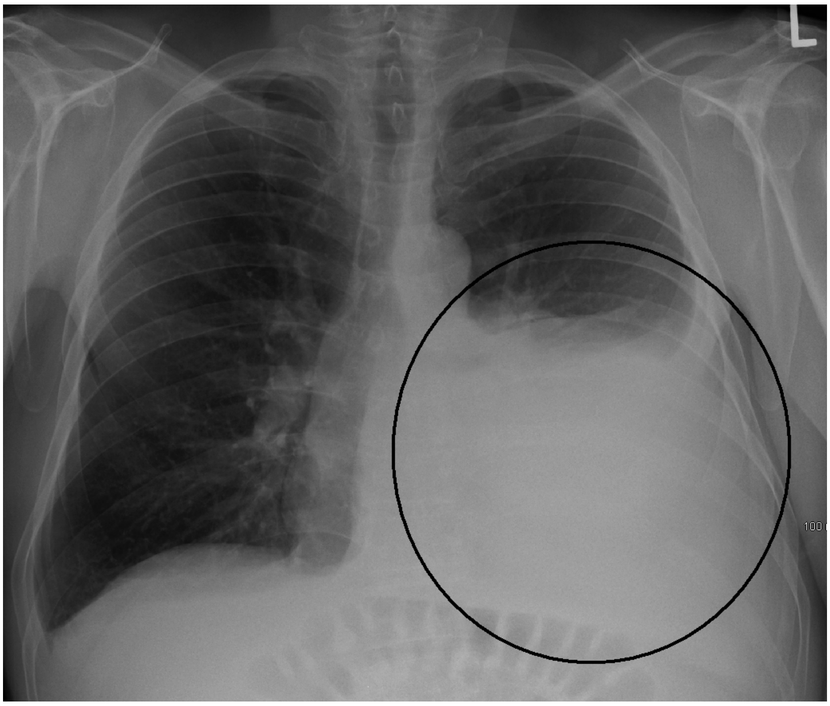

Figure 1.4: A large left sided pleural effusion. Author: James Heilman, MD.

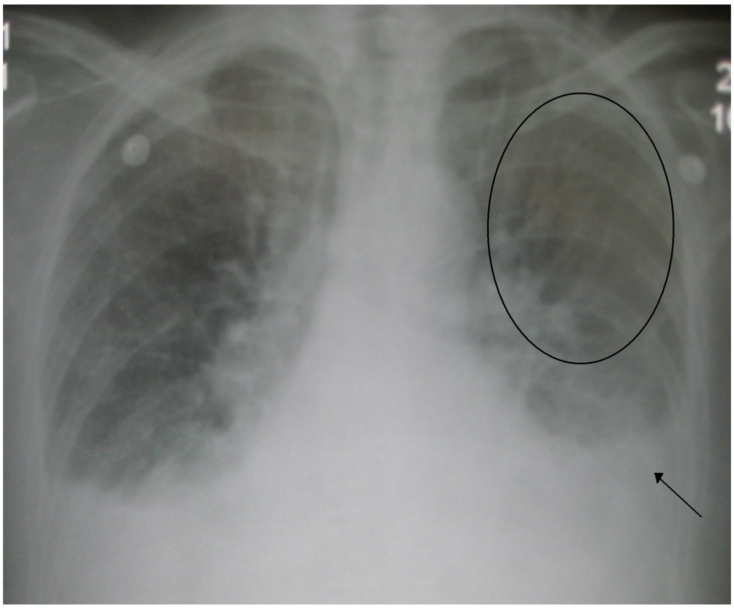

Figure 1.5: Heart failure: Enlarged heart size, atypical vascular redistribution (circle) and small bilateral pleural effusion (arrow). Author: James Heilman, MD.

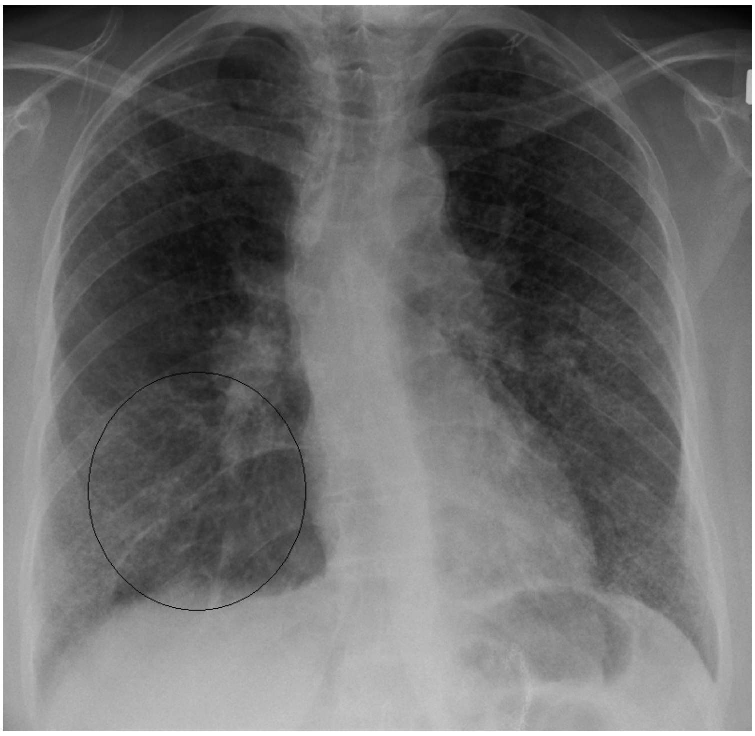

Figure 1.6: Typical nodularity with sarcoidosis. Author: James Heilman, MD.

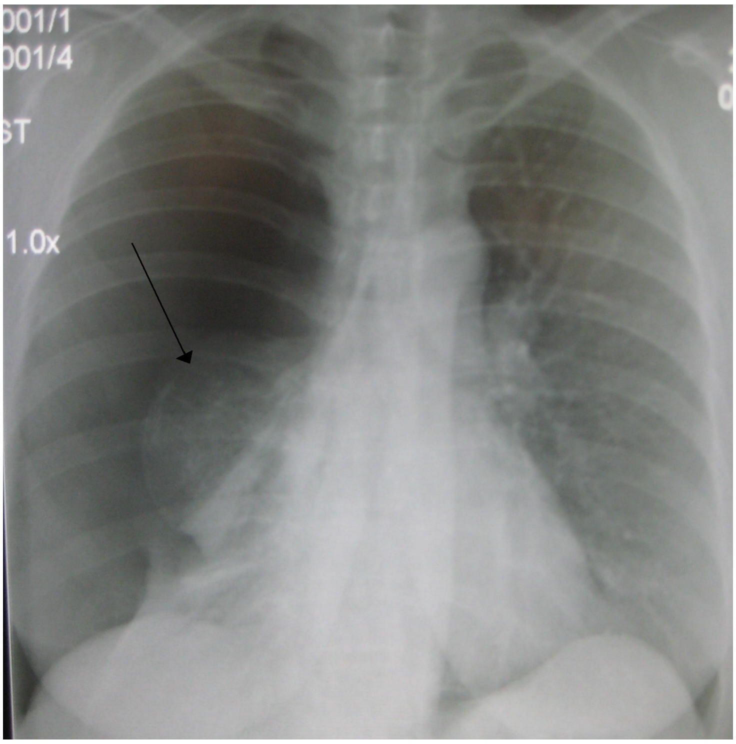

Figure 1.7: A large right sided spontaneous pneumotorax. An arrow indicates the visible edge of the collapsed lung. Author: James Heilman, MD.

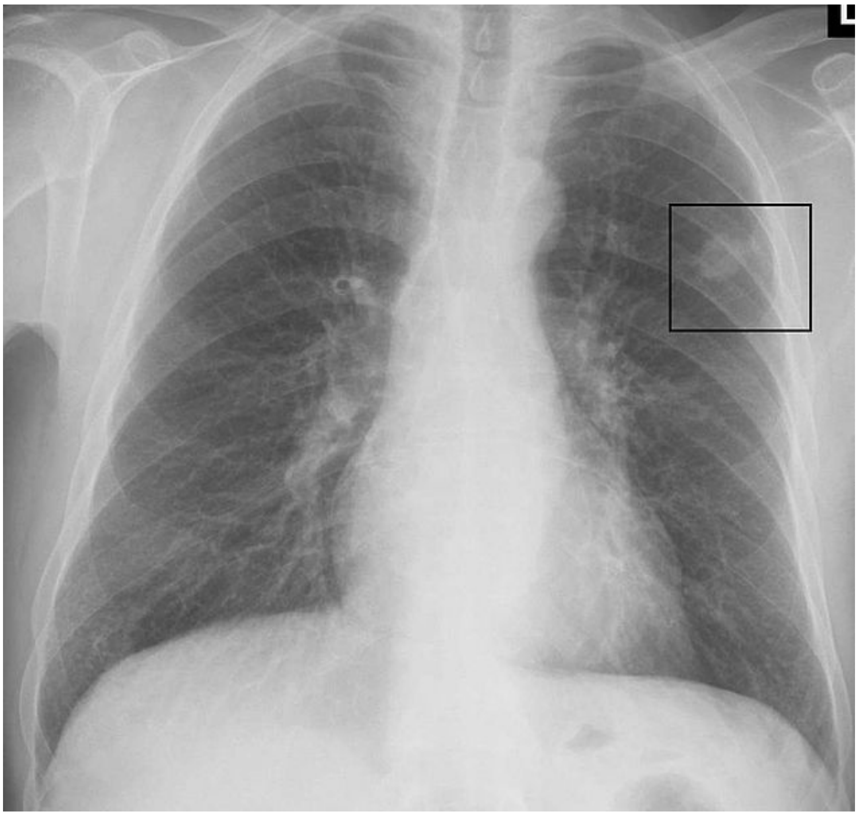

Figure 1.8: Tumor on the upper left side of the chest.

Chapter 2

Alignment

2.1 What do we pretend?

Chest x-ray images are hard to compare using computer vision techniques. Every chest has its own shape. Most of the symptoms a doctor can find in this kind of images are located in the lungs region. Lung segmentation is a necessary step to make easy the location of the abnormal regions, lungs look much more uniform than other regions, so it will be easy to find strange regions.

Segmentation is an obvious previous step to extract features and apply some machine learning techniques because it transforms the problem to an easier learning problem where variability is smaller. But taking in to account the features that we pretend to extract after segmentation, medical science suggest us that position is an important feature that doctors consider when they decide to suspect that a region is abnormal. For example, cancer is more likely in some lungs regions than others.

We pretend to align all the lungs to a chest template so the exact location of each pixel inside the ribs cage is known. Pixel location will be used to predict doctors eyes gaze. Once aligned, segmentation will be a trivial task. Alignment process must preserve as much as possible the image proprieties, specially the ones useful for doctors to detect symptoms. Thin plate splines is the technique used to distort the images to align them using some landmark points.

2.2 Thin plate splines

Given a set of control points $\{(x_i, y_i), i = 1, 2, \ldots, K\}$, we pretend to interpolate the function $f : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ that fits them imposing some smooth conditions on it, so $f(x_i) \approx y_i$. We consider radial basis functions $\phi(r) = e^{-r^2/\sigma^2}$ to approximate f.

$$f\left(y\right)=\sum_{i=1}^{K}c_i\cdot\phi\left(\left\|y-x_i\right\|\right) \tag{2.1}$$

Softness and error measure of f are needed in order to find the parameters that optimize those magnitudes. Second derivatives respect x and y define a good measure of smoothness with the energy function shown on 2.2.

$$M_{soft}\left(f\right)=\int\left[\left(\frac{\partial^2 f}{\partial x^2}\right)^2+\left(\frac{\partial^2 f}{\partial x\partial y}\right)^2+\left(\frac{\partial^2 f}{\partial y^2}\right)^2\right]dxdy \tag{2.2}$$

Error measure used is the typical squared error as shown in 8.1.

$$M_{error}\left(f\right)=\sum_{i+1}^{K}\left\|y_i-x_i\right\|^2 \tag{2.3}$$

Finally, we need to minimize the objective function $M=M_{soft}+\lambda\cdot M_{error}$ respect f, where λ measures how important is the error respect the softness measure.

$$f=argmin_f M\left(f\right) \tag{2.4}$$

2.3 Building the chest template using landmark points

Control Points (landmarks) are necessary to apply thin plate splines for chest alignment. Since we only have 100 images, manual landmark selection is possible. Automatic landmark finding is a solved problem of image processing, algorithms such as Active Shape Models could be used in a final user application. Ribs and the spine are the perfect landmarks for our proposes. The 30 points at 2.1 were selected to be the references of the alignment process.

1. Intersection between 1st and 2nd right ribs.
   ...
8. Intersection between 8th and 9th right ribs.
9. Joint point of 1st right rib and the spine.
   ...
15. Joint point of 7th right rib and the spine.
16. Intersection between 1st and 2nd left ribs.
   ...

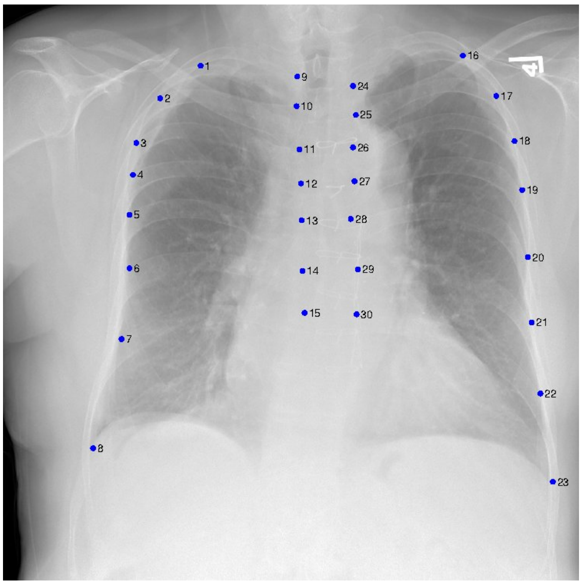

Figure 2.1: Chest Model landmarks

23. Intersection between 8th and 9th left ribs.

24. Joint point of 1st left rib and the spine.

    ...

30. Joint point of 7th left rib and the spine.

Once all landmarks are identified, the mean of all chest profiles is computed to obtain the landmarks of a mean chest that will be our chest model of reference to align all our set of images. Chest model is computed as shown in 2.5 where N is the number of images and $i = 1 \dots 30$ is the number of landmark.

$$(x_i, y_i) = \frac{1}{N} \sum_{k=1}^{N} \left(x_i^k, y_i^k\right) \tag{2.5}$$

Figure 2.2 shows the chest template in red and the original chest points in blue. And figure 2.3 the result of aligning the original points to the template. As it is clear in figure 2.3, there has been a deformation to make the landmarks fit. However, objects are as clearly visible in the aligned image as the original. The deformation should not affect to the mass detection process.

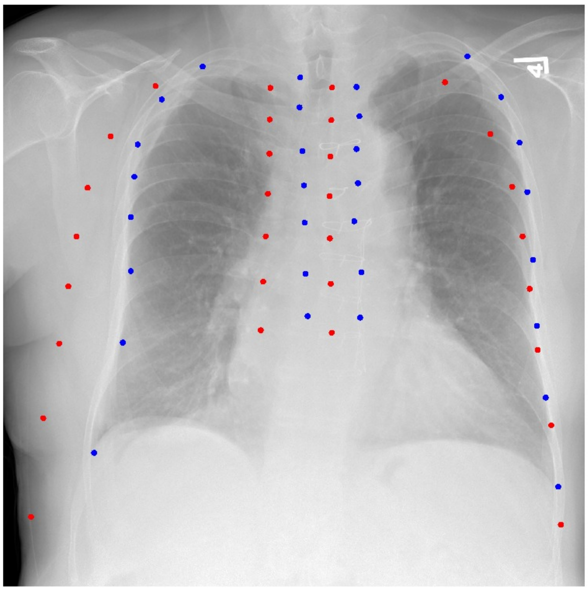

Figure 2.2: Original Chest (blue) vs chest template points (red)

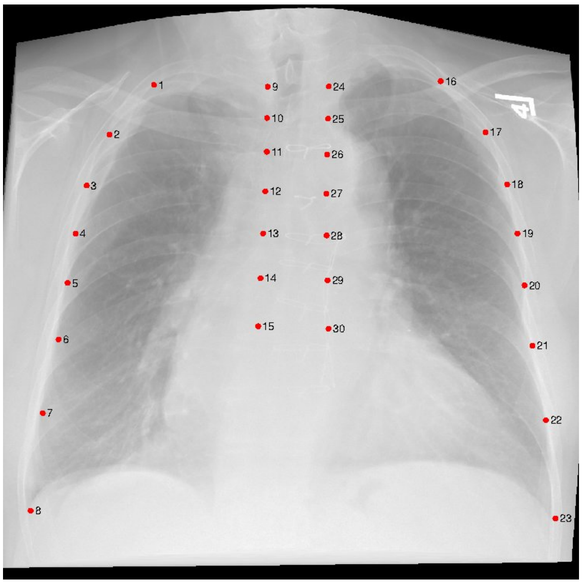

Figure 2.3: Chest template points coincide with the landmarks of the aligned image (red)

Chapter 3

Convergence index filters for lung nodule searching

3.1 Introduction to convergence index filters

Convergence index filters is an image processing technique that pretends to measure the gradient converge to each pixel in a neighbourhood around it. That is interesting for us because could be useful to find nodules in the image, which could be a symptom of cancer. Since CIF measure the convergence to a central point, this algorithm tends to detect rounded bright areas. The size of the neighbourhood must be similar to the size of the nodule detected.

3.2 General convergence index filter

Prewitt gradient is used as an estimation of partial derivatives of the image value 3.3.

$$\frac{\partial f}{\partial x} \approx g_x(x,y) = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} * f(x,y) \tag{3.1}$$

$$\frac{\partial f}{\partial y} \approx g_y(x,y) = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} * f(x,y) \tag{3.2}$$

$$g(x,y) = [g_x(x,y), g_y(x,y)] \tag{3.3}$$

Once defined the gradient we are going to use, the output of the filter is computed as shown in 3.4, where N is a neighbourhood around P. The integrand at 3.4 is exactly the cosine of the angle between $\overrightarrow{PQ}$ and $g\left(Q\right)$. The maximum convergence is produced when the cosine reaches his maximum. So what is computed at 3.4 is an average of the convergence in the neighbourhood of P, N.

$$C(P) = \frac{1}{\mu\left(N\right)} \int\limits_{N} \frac{\overrightarrow{PQ} \cdot g\left(Q\right)}{\|\overrightarrow{PQ}\| \cdot \|g\left(Q\right)\|} \tag{3.4}$$

Over that general description of a CIF, several modifications can be done. First, many different neighbourhoods can be defined to be the regions of support around a pixel.. Second, for computational reasons, not all pixels in each region can be taken in to account in 3.4. That is why is usual to take some of the pixels in a region. Last, computing a cosine for each pixel in each neighbourhood is a very time consuming process, that will be solved with a previously computed table of cosines, as it is shown in the following sections.

3.3 Coin Filter

Coin filter is a particular case of a convergence index filter where only a small set of points inside a spherical neighbourhood are taken in to account at 3.4. The procedure to select those points is composed by two steps. First, selecting N equispaced radial lines from the central pixel to R_{max}, the end of the region. Second, for each line i we take the average from the central point to Q_i. Where Q_i is the point on the line that maximize the average in 3.5. Finally, the average of those points selected in 3.7. As a theoretical approach, input image $f\left(x, y\right)$ can be considered a continuous function. However, when coin filter is implemented, integral compute for each line will be substituted by an averaged sum for some pixels on the line, as it will be shown later.

$$C_i\left(x, y\right) = \max_{Q_i} \left\{ \frac{1}{\|P - Q_i\|} \int_{P}^{Q_i} \cos\phi\left(l\right) dl \right\} \tag{3.5}$$

$$0 < \|P - Q_i\| < R_{max} \tag{3.6}$$

$$C\left(x, y\right) = \frac{1}{N} \sum_{i=1}^{N} C_i\left(x, y\right) \tag{3.7}$$

3.4 Sliding Band Filter (SBF)

SBF is a more sophisticated version of CIF. This modification tries to find the edges of bright areas using the gradient of the edges which is less noisy than center. Noisy

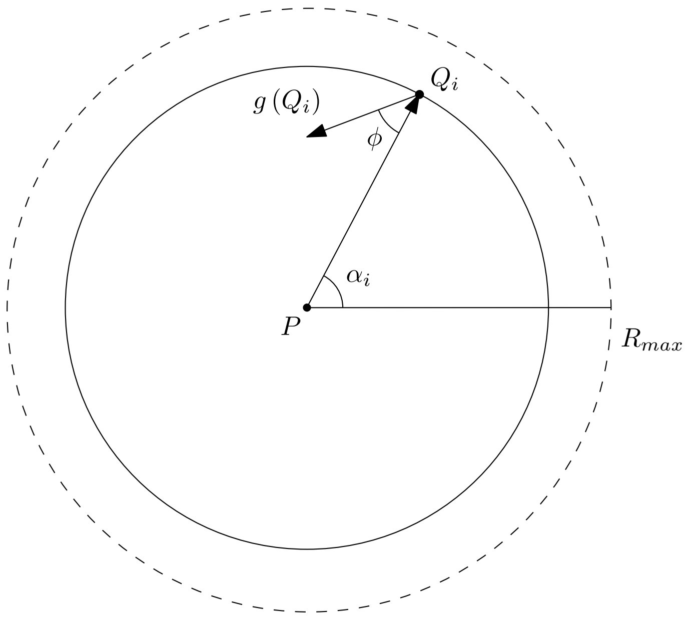

Figure 3.1: Coin filter

gradient appears in the center of nodules and also in flat areas with no bright objects.

$$C_i(x,y) = \max_{Q_i}\left\{\frac{1}{d}\int_{Q_i}^{Q_i+d\cdot\vec{r}} \cos\phi(l)\,dl\right\} \tag{3.8}$$

$$R_{min} < \|P - Q_i\| < \|P - (Q_i + d\cdot\vec{r})\| < R_{max} \tag{3.9}$$

3.5 Computational issues

Three important aspects must be taken in to account, in order to obtain the Slided Band Filter fast of a discrete image.

First, as we have said before, we are not computing the value of the *cos* $(.\because.)$ for each pixel of the neighbourhood. Tanking some equispaced angles of a circle centred on the pixel and some steps along the radial lines on that angles is enough to estimate the convergence to the center.

Second, since the image is a discrete signal, 3.8 and 3.5 must be approximated using a sum of the value of the pixels along (interpolated by nearest pixel) the line instead of an integral.

And last, the most time consuming operation of the process is computing the cosine. In order to do it fast, a pre-computed table of cosine values is used. Then, once the value

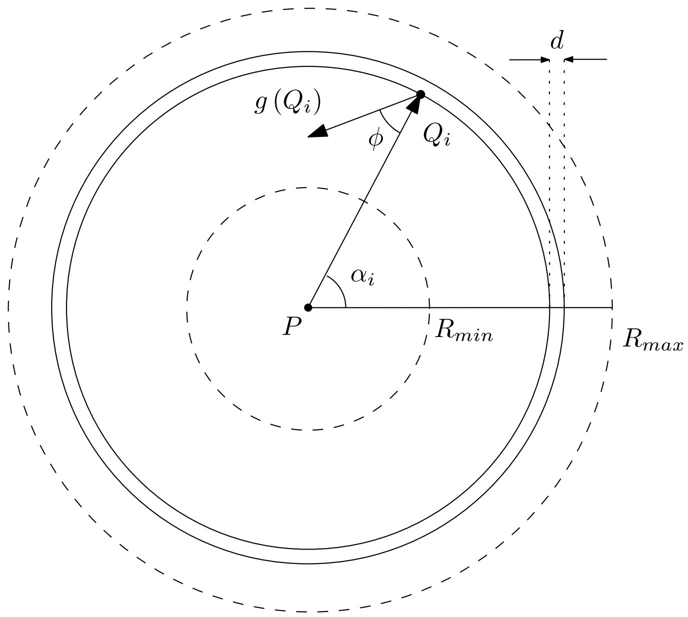

Figure 3.2: Slided Band Filter

of ϕ is obtained, the cosine of the nearest angle of the table is used as an approximation of $cos\,(\phi)$.

3.6 Examples

This section shows some examples of the Slided Band Filter performance. 3.3, 3.4 and 3.5 show an artificial example of the detection of a bright rounded region on a dark background with some noise on it. It is an easy example but shows how SBF works.

Figure 3.5 shows the image at 3.4 after being filtered. High output values hare found around the center of the image, showing the convergence of the gradient to the center. Lower values far from the center correspond to a more flat region of the image. The random values at the image borders correspond to small random gradient convergences of the gaussian noise added to the image.

Next examples is a real case. A suspicious region is visible between the 7th and the 8th left rib is visible at 3.6. At 3.7 the slided band filtered output is shown. Clearly, the suspicious mass has produced a bright spot. However, near the spine some other bright regions have appeared. Looking again at 3.6 is it clear that those bright spots correspond to de bronchioles that, as seen in BLABLA, usually appear as bright regions in chest x-ray images.

Figure 3.3: Bright round region on a dark background

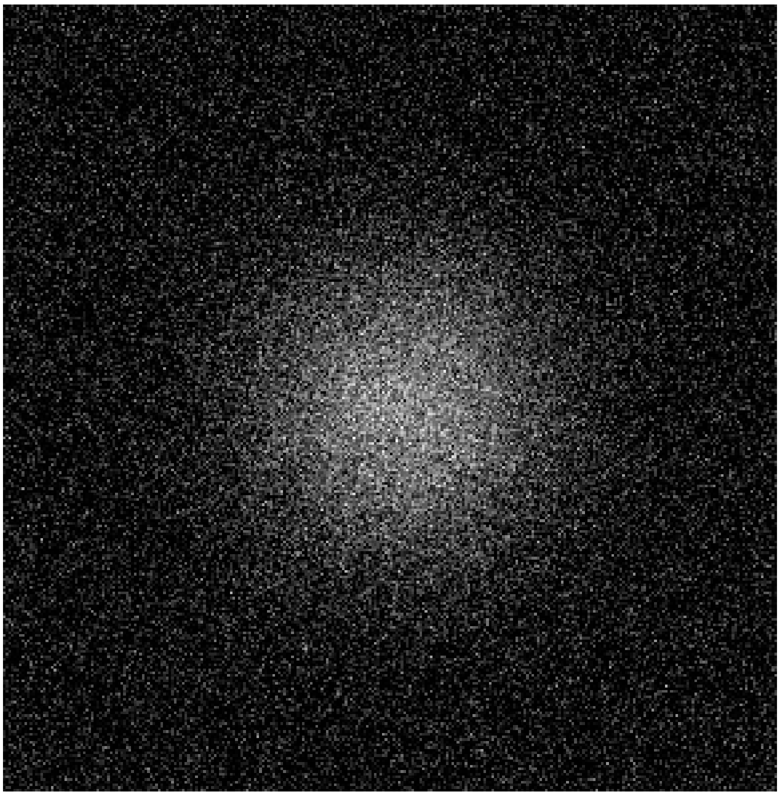

Figure 3.4: Gaussian noise on a bright round region

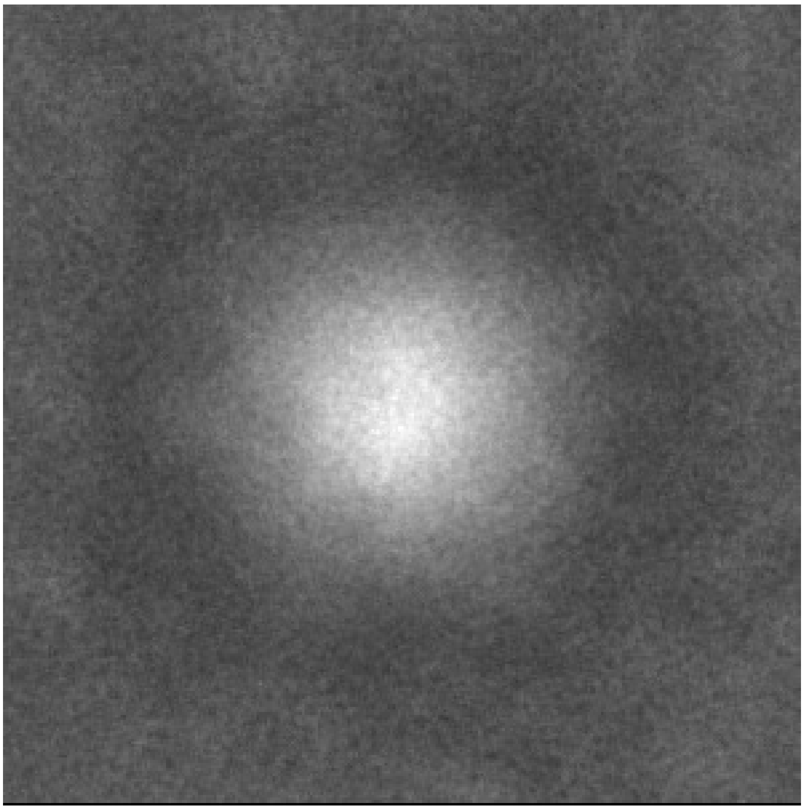

Figure 3.5: SBF output of the noisy bright region

Figure 3.6: Chest x-ray image with a suspicious mass

Figure 3.7: SBF output of 3.6

Chapter 4

Tuning the Sliding Band Filter

This chapter goal is understanding SBF parameters impact on the filter output in order to get the best configuration for nodule detecting on x-ray images. Preprocessing the images is needed to get interesting results, the way of doing it is also considered in this chapter. Finally a post processing method is presented to improve the results of the SBF algorithm.

4.1 Pre-processing the images before Sliding Band Filtering

High frequency components make gradient very rough. Small image value variations determine the direction of the gradient at each point. In order to detect relatively big nodules at the image, those high frequency components must be eliminated. Low pass filter has to be applied before gradient convergence is measured.

Obviously, high cutoff frequency forces gradient convergence to be determined by smaller objects and a low cutoff frequency makes bigger objects determining. As a consequence of that, it is a trade-of between big and small nodules detection when the size of the low pass filter is chosen.

4.2 Tuning SBF parameters

SBF parameters have been introduced in the last chapter:

1. Maximum radius R_{max}
2. Minimum radius R_{min}
3. Band size d

Other parameters could be considered, such as how many radial lines use to estimate the gradient convergence, the size of the cosine table used to compute $cos\,(\phi)$, etc. Those parameters are not taken in to account in this section because are not relevant for the size of the nodules that SBF can detect. Number of lines is set to $N = 32$ such as the size of the cosine table.

Detectable nodules average radius is between R_{min} and R_{max}. The value selected for this two parameters must be consistent with the low pass filter size applied during the preprocessing stage. The smoothing filter will never eliminate high frequency components enough to set $R_{min} = 1$ and R_{max} at some big value in order to cover all the possible nodule sizes. So, relatively near values must be chosen for those parameters.

Finally, d determines how pronounced is the edge of the nodules detected. Typically, big nodules tend to look like big subtle shadow with a smooth edge. That is why, in order to detect big suspicious areas, a big value of d is needed.

4.3 Multi-resolution approach to nodule detection

A high parameter impact on the size of detectable nodules has been observed in the last sections. This behaviour suggests us that different SBF configurations should be used to detect different kind of nodules. Despite that option would work, SBF is a very time consuming process when the image is big, doing it multiple times for each image is too much. Instead of using different parameters, a multi-resolution process has been chosen. Full size image is not needed if a low pass filter is going to be applied as a preprocessing step to de SBF. Down-sampling the image after applying an anti-aliasing filter is enough to detect bigger suspicious areas in the image. Finally, we can apply the same configuration for the parameters R_{min}, R_{max} and d, different image sizes allow us to detect different size nodules.

As an example, we are going to use the image 4.1. Different objects are visible for us, but it is important to realize that there is a big white shadow around the 7th right rib. Using the multi-resolution SBF method, this shadow is clearly visible at 4.2, extracted with the smallest image. As image size is increased, new possible nodules appear at 4.3 and 4.4. For example, the small rounded area in the 4th right rib, near the spine. But, if the size is increased the big white shadows is not detectable any more.

4.4 Highlighting the most probable gazed regions based on position

Introduction chapter describes how bronchioles can be easily confused by suspicious nodules. Specially because the rounded shape of both objects. Taking in to account

Figure 4.1: Example of chest x-ray image before multi-resolution SBF processing

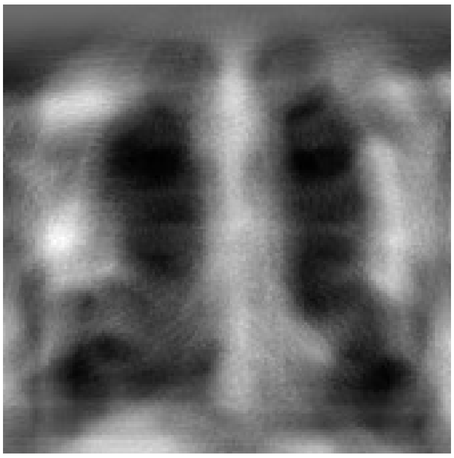

Figure 4.2: SBF output. $R_{max} = 25$, $R_{min} = 10$, $d = 8$, $resolution = 169x169$

that SBF measures the gradient convergent regions (witch means rounded areas), bronchioles and other rounded areas have similar SBF outputs.

The reason why doctors know that bronchioles are not tumours is that cancer tends to appear far from and the bronchioles near the spine. During the training process followed by a human to be able to read x-ray images, doctors learn an estimation of the probability of finding a nodule indicating cancer or other diseases at each part of the lung. We pretend to imitate this learning process using the dataset of the gazed points on the x-ray images we have. In this section we are going estimate the probability density function of a point of being gazed given only the position: $P(x, y)$ where x and y are the coordinates of the point. This information is going to be used to highlight the regions where cancer usually appears (so doctors tend to look at) of the SBF output.

To estimate $P(x, y)$, we are going to calculate the two dimensional histogram matrix

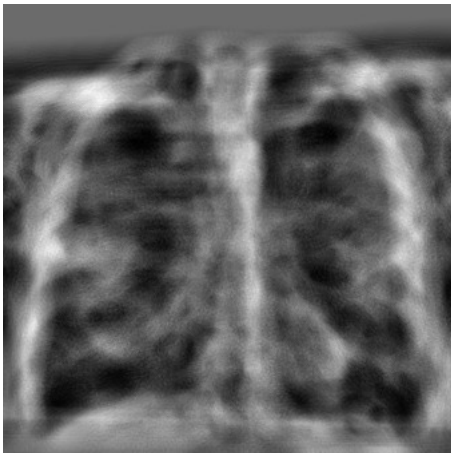

Figure 4.3: SBF output. $R_{max} = 25$, $R_{min} = 10$, $d = 8$, $resolution = 337x337$

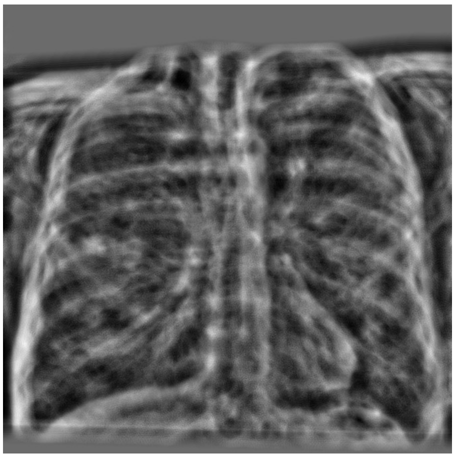

Figure 4.4: SBF output. $R_{max} = 25$, $R_{min} = 10$, $d = 8$, $resolution = 674x674$

$H_{N\times N}$ of the gazed points position. N corresponds to the number of bins per dimension. Each element of H, $h_{i,j}$ is the number of samples of the bin $\{i, j\}$. In our case, we have used a subset of the images we have (it will correspond to the training set used in the next chapters for the relevance vector machine training presented there) to compute this histogram using $N = 20$, see figure 4.5. Probability density function is obtained interpolating the histogram using bi-cubic interpolation and normalizing the result. Finally, we obtain a $2022x2022$ matrix, witch is the size of the images we have, with the estimated probability of being gazed for each pixel 4.6.

Probability density function estimated gives us information of what regions could likely present cancer. Weighting the output of the SBF filter using the estimated probability, really suspicious nodules (because are located in a frequently suspicious area) are going to be highlighted but not the regular ones. This is specially important

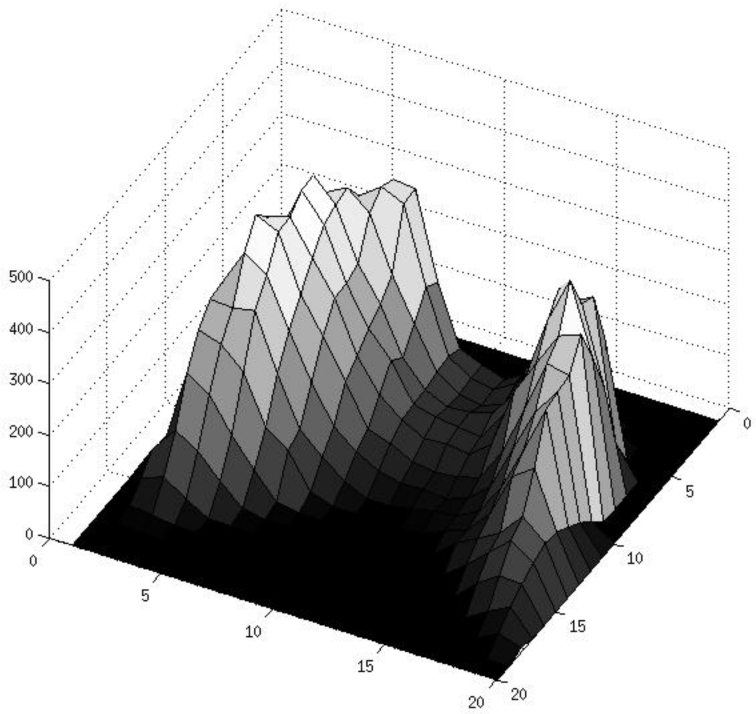

Figure 4.5: Histogram of gazed points positions

to avoid confusions with bronchioles. Results for the particular example of last section are shown at 4.7.

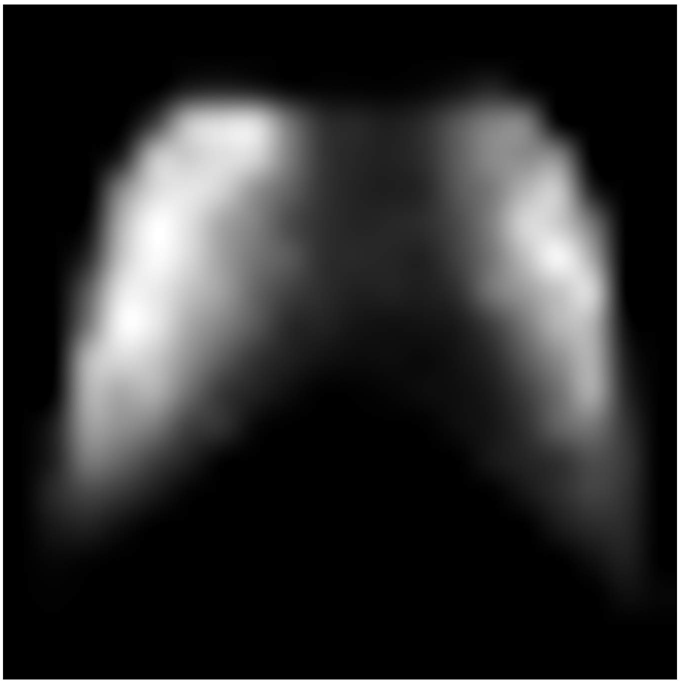

Figure 4.6: Probability density function of gazed points position

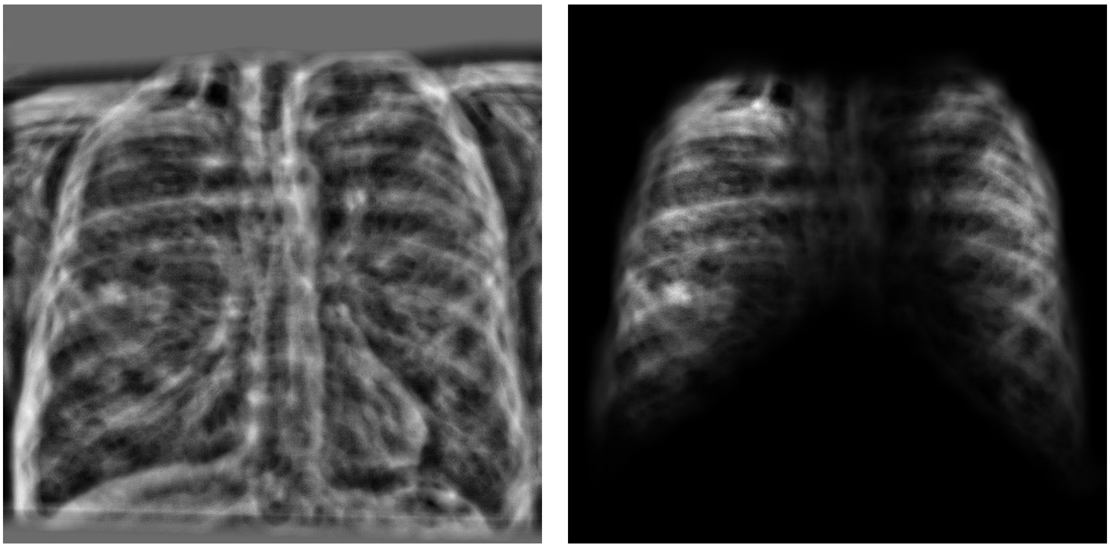

Figure 4.7: SBF output before (left) and after being weighted by gazed points position PDF (right)

Chapter 5

Using watershed transformation for nodules candidates segmentation

5.1 Introduction to watershed

The output of the Slided Band Filter shows the high convergent areas as the brighter areas of the image. In order to apply Relevance Vectors Machine to train an algorithm to identify suspicious areas we need to segment the image into different regions of interest where a nodule could be found. Watershed transformation is the technique used for segmentation.

Considering an image as a surface at $\mathbb{R}^3$ where the x and y are the position of the pixel and the value $f(x, y)$ is the z axis value. We can consider two kinds of points:

1. Points where a water drop would fall in to a minimum M_k (not equilibrium points). Also called watershed points of M_k.

2. Points where a water drop could fall in to one minimum or other (unstable equilibrium in at least one direction). Also called watershed lines

Watershed algorithm is able to identify what points will fall to each minimum and what are the watershed lines, where is not clear where would fall.

5.2 Watershed algorithm

Watershed algorithm is a recursive process that pretends to emulate the flooding process of the $\mathbb{R}^3$ surface presented in the last section. To get started, some definitions must be done. Let $M_1, \cdots M_R$ be the local minimums of the image $f(x, y)$ and $C(M_i)$ the watershed points of M_i. We define $T[n]$ as the points of the image with a value lower than n, 5.2.

$$T[n] = \{(s, t) \,|\, f(s, t) < n\} \tag{5.1}$$

Also, let $C_n(M_i)$ be watershed points of M_i with values lower than n, 5.2. The union of the watershed points for all the minimums M_i is defined as $C[n]$, 5.2. Finally, let $Q[n]$ be set of connected components of $T[n]$.

$$C_n(M_i) = C(M_i) \cap T[n] \tag{5.2}$$

$$C[n] = \cup_{i=1}^{R} C(M_i) \tag{5.3}$$

The algorithm is initialized with $C[\min\{f\} + 1] = T[\min\{f\} + 1]$. At each step n, we suppose that $C[n-1]$ is correctly constructed and for each $q \in Q[n]$ there are three possibilities:

1. $q \cap C[n-1]$ is empty.

2. $q \cap C[n-1]$ contains one connected component of $C[n-1]$.

3. $q \cap C[n-1]$ contains more than one connected component of $C[n-1]$.

In the first case, q is watershed region associated to a new local minimum. In this case, q is added to $C[n]$. In the second case, q is contained in the watershed region associated a one minimum, so is added to. In the last case, two watershed regions associated to a two or more different minimums have become a single connected region, q, so we need to find the pixels where this regions are touching.

To do so, we consider the connected components of $C[n-1]$ contained in q, A and B. If more than two components were contained in q, the process would be the same. First, both regions must be dilated by structuring element e 5.2

$$e = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \tag{5.4}$$

In the figure 5.2 is shown two steps of the region growing process. New pixels can not be in touch with both regions, pixels of the dilation that are in both connected component are not added to $C\left[n\right]$. Those pixels form the watershed lines, boundaries of watershed regions. In order to keep the watershed regions separated, a dam is build on those pixels. Normally the value of those pixels is set to $max\{f\}+1$. Once $n = max\{f\}$, the algorithm finishes and each connected component of $C\left[n\right]$ is a watershed region. Also, the pixels in $\{(x,y)\,|f\,(x,y) = max\{f\}+1\}$ correspond to the watershed lines.

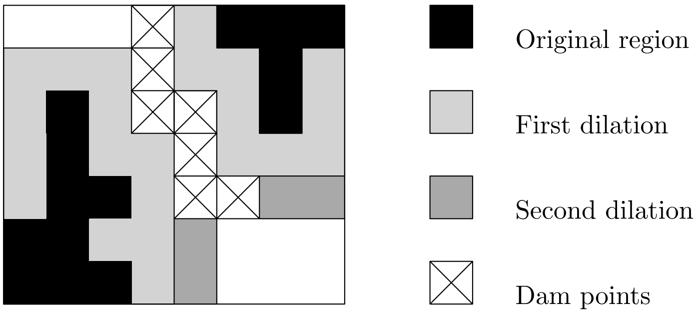

Figure 5.1: Region growing

5.3 Segmenting the Slided Band Filtered images

Watershed has been used to segment output of the Sliding Band Filter in order to obtain different regions of interest. Our intention is to separate all possible suspicious nodules. As you may have noticed, watershed as it is explained in the last section does exactly the opposite: the regions segmented are low valued regions of the image. That is why the image must be inverted before applying watershed, as shown in 5.3. Generally, the gradient of SBG images could be very noisy. In our case, we are only interested in segment relatively big areas, we are talking about hundreds of pixels. Our algorithm should not take in to account small watershed regions. To do so, a low pass filter is used on SBF images to make the their gradient softer. We are using a Gaussian filter of 50 pixels with a variance of 30 pixels.

$$f_{out}\left(x,y\right) = -f_{in}\left(x,y\right) + max\{f_{in}\left(x,y\right)\} \tag{5.5}$$

Given an image to witch an Sliding Band Filter has been applied such as 5.2, the watershed segmentation is able to separate the lungs in regions of interest, bright areas of the SBF output or gradient convergent regions of the original image. Those results are shown in the figure 5.2. As it is explained in following chapters, this segmentation is a key step in our process. Each region is going to be considered an input sample for a statistical learning algorithm. This machine learning technique is introduced in the following chapters.

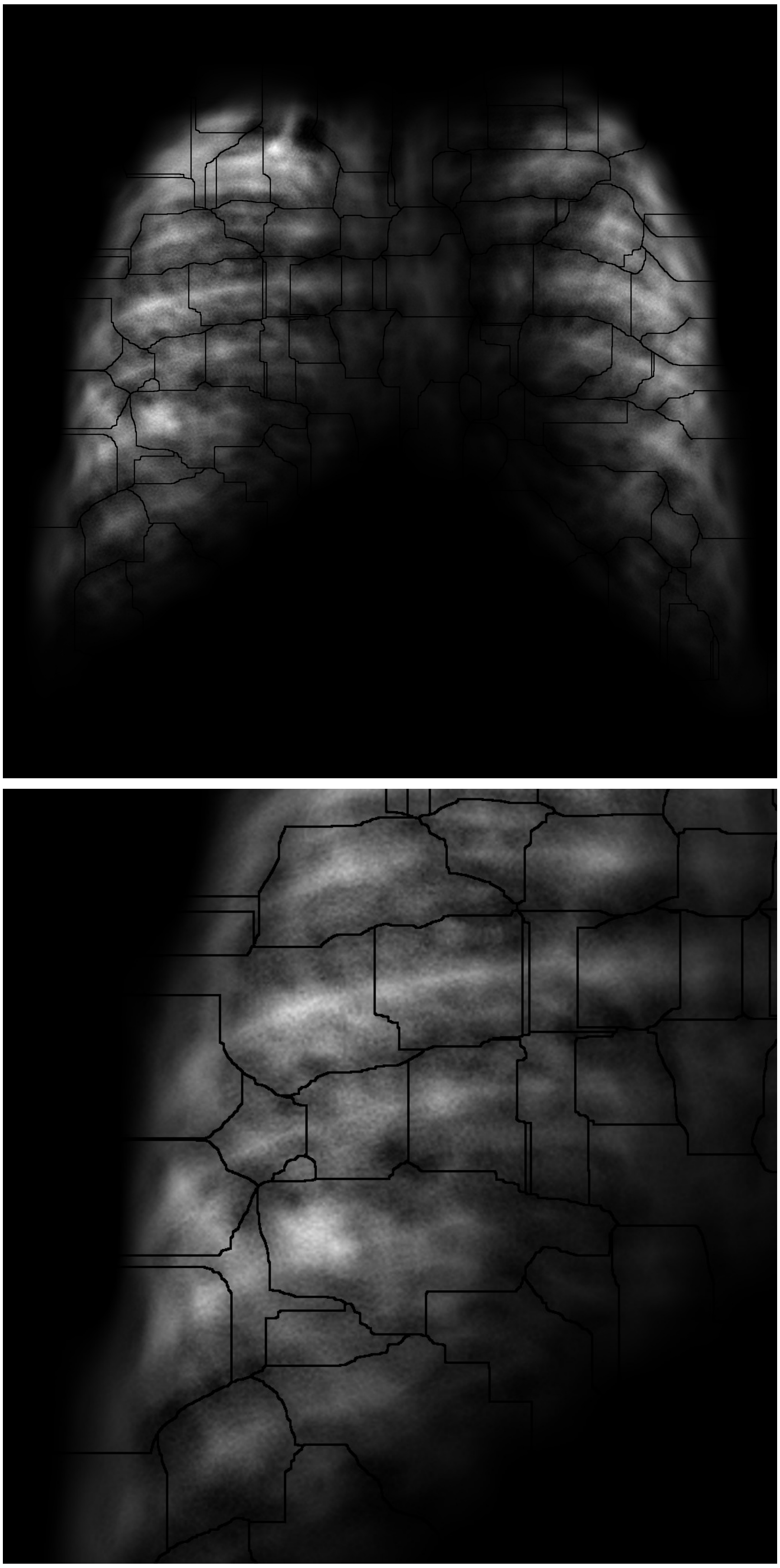

Figure 5.2: Convergence map segmented using watershed

Chapter 6

Feature selection

6.1 Chosen features

We pretend to apply machine learning to obtain an algorithm able to predict the gazed regions of a x-ray image. The procedures shown on the first chapters show pre-processing algorithms. In the case of the watershed algorithm, each region segmented represents a sample of our statistical learning method. Using the outputs of the Sliding Band Filter, we expect to find features correlated with the number of gazed points per region. In this section some reasonable features are suggested, next section presents a method to reduce feature space dimensionality using principal component analysis.

SBF output tends to measure the gradient convergence at each pixel. Given one nodule candidate, a sample of region segmented with watershed, next values makes sense be taken in to account as features:

1. Mean convergence index.

2. Convergence index variance.

3. Maximum convergence in the sample.

4. Minimum convergence in the sample.

5. Size of the region sample (in pixels).

6. Perimeter of the region sample.

7. Eccentricity of the region sample.

8. Centroid coordinates of the sample (x, y).

Usually, doctors use the symmetry of the human body to detect differences between left and right lungs. That indicates us that convergence features on the symmetric region of the chest could be an interesting feature. So given a region, we can compute its symmetric region and extract same features again as a features of our initial sample.

Since we are using a multi-resolution algorithm to extract de SBF, first 4 features of the list above can be extracted for each resolution level and for the symmetric region. Finally, we are going to work with three resolution levels: $2 \cdot 3 \cdot 4 + 5 = 29$ features.

In the next section, this big number of features will be reduced using PCA algorithm.

6.2 Principal Component Analysis

We pretend to reduce the number of features obtaining the most representative set. That means that we want to work with features that highly discriminate our samples. In other words, those features must have the highest variance possible, computed using the samples we have.

Principal Component Analysis (PCA) can project our data in a smaller space of features that discriminates better the samples. Given a features matrix X where each row is a sample and each column a feature, we compute the covariance matrix 6.2 and its eighenvectors and eighenvalues. Since we want a high variance for each feature, we pick the eighenvectors witch eighenvalues are higher being $\mathbf{A}$ a matrix containing them. Eighenvectors in $\mathbf{A}$ must be normalized. The dimension of the resulting feature space is set by the number of eighenvectors chosen.

$$\boldsymbol{\Sigma} = (\mathbf{X} - \boldsymbol{\mu})^T (\mathbf{X} - \boldsymbol{\mu}) \tag{6.1}$$

Now we have decided the subspace of the features space where we are going to project each sample. Let $\mathbf{x}_i$ be the ith sample in $\mathbf{X}$, a row vector. Our new sample is the projection of $\mathbf{x}$ in the subspace defined by $\mathbf{A}$ 6.2. Finally, the output sample set is given by 6.2 where N is the number of samples.

$$\mathbf{y}_i = \mathbf{A}^T \mathbf{x}_i^T \tag{6.2}$$

$$\mathbf{Y} = (\mathbf{y}_1, \cdots, \mathbf{y}_N)^T \tag{6.3}$$

Chapter 7

Sparse Bayesian Learning

7.1 Introduction

Given a set of examples of inputs vectors $\{\mathbf{x}_n\}_{n=1}^{N}$ and output targets $\{t_n\}_{n=1}^{N}$, we wish to learn a model of the dependency of the targets on the inputs. Using that model we would be able to make predictions of t for a given sample $\mathbf{x}$. Real world data examples are noisy. That means that small errors are expectable on the data. Reasonable model would have an error, otherwise over-fitting phenomena is occurring so the model has not generalization capacity. Usually, $\{\mathbf{x}_n\}_{n=1}^{N}$ are called features vectors because.

Usually, the way of seeing the problem is considering a target function 7.1 that our algorithm needs to learn. $\mathbf{w} = (w_1, w_2, \cdots, _N)^T$ are the parameters to learn and $\phi : \mathbb{R}^M \to \mathbb{R}^N$ a fixed non linear map of the M-dimensional space of features $\phi(\mathbf{x}) = (\phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \cdots, \phi_N(\mathbf{x}))^T$. Functions ϕ_i are also called basis functions.

$$y(\mathbf{x}; \mathbf{w}) = \mathbf{w}^T \cdot \phi(\mathbf{x}) \tag{7.1}$$

Using $\phi_i(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}_i)$ is a particular case of 7.1. For each sample of the training set $\mathbf{x}_i$ a basis function is constructed with the kernel function K. Predicting based on functions like 7.2 is what Support Vector Machines do. SVM is a very extended machine learning technique because in classification applications is able to minimize the error of prediction and maximize the margin between two classes. This last property is very appreciated for overfitting avoiding.

$$y(\mathbf{x}; \mathbf{w}) = \sum_{i=1}^{N} w_i \cdot K(\mathbf{x}, \mathbf{x}_i) + w_0 \tag{7.2}$$

$$p(\alpha) = \prod_{i=0}^{N} Gamma(\alpha_i | a, b) \tag{7.3}$$

Despite the success of SVM, this technique is not a probabilistic method. Ideally, we would like to predict a target t given a feature vector sample $\mathbf{x}$ estimating the conditional distribution $p(t|\mathbf{x})$ in order to capture uncertainty of the prediction. That is the main reason why we are introducing a less known treatment of the problem called Relevance Vector Machine. RVM is a Bayesian treatment to estimate the parameters $\mathbf{w}$ of 7.2.

7.2 Model Specification

Lets consider a data set of input (features) and targets $\{\mathbf{x}_n, t_n\}$ for $n = 1$. Our goal is tu estimate the parameters of $y(\mathbf{x}_n; \mathbf{w})$ assuming the model shown in 7.4. Where ϵ_n are samples of a zero-mean Gaussian noise process with variance σ^2 and y responds to the approach shown in 7.2. Applying this statistical model, posterior probability density function 7.5 is obtained. Remember that $\mathbf{t} = (t_1, \cdots, t_N)^T$, $\mathbf{w} = (w_0, \cdots, w_N)^T$ and $\mathbf{\Phi}$ is the $N \times (N+1)$ design matrix with $\mathbf{\Phi} = [\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \cdots, \phi(\mathbf{x}_N)]^T$ and $\phi(\mathbf{x}_n) = [1, K(\mathbf{x}_n, \mathbf{x}_1), K(\mathbf{x}_n, \mathbf{x}_2), \cdots, K(\mathbf{x}_n, \mathbf{x}_N)]^T$.

$$t_n = y(\mathbf{x}_n; \mathbf{w}) + \epsilon_n \tag{7.4}$$

$$p\left(\mathbf{t}|\mathbf{w}, \sigma^2\right) = \left(2\pi\sigma^2\right)^{\frac{-N}{2}} \cdot \exp\left\{\frac{-1}{2\sigma^2}\|\mathbf{t} - \mathbf{\Phi}\mathbf{w}\|^2\right\} \tag{7.5}$$

Given a model like 7.5, it would be a common solution tu apply maximum likelihood over the parameters $\mathbf{w}$ and σ^2. However, this procedure would lead us to an over-fitting phenomena, obtaining a very good parameter values but just for our training data set. At this point, SVM maximizes the margin of decision to avoid over-fitting. Since RVM pretends to get a probabilistic alternative, it is necessary to impose a prior probability distribution over the parameters. We impose a smooth function by choosing $\mathbf{w}$ as a zero-mean Gaussian distribution in 7.6. The $N+1$ new parameters α, usually called hiperparameters, need to be constricted to a hiperprior uniform distribution, the same as $\beta = \sigma^{-2}$.

$$p(\mathbf{w}|\alpha) = \prod_{i=0}^{N} \mathcal{N}\left(w_i|0, \alpha_i^{-1}\right) \tag{7.6}$$

7.3 Inference

Our problem is making predictions of the target t_* given a test point $\mathbf{x}_*$ in terms of the predictive distribution 7.7, the probability of the prediction given the training set. What we need to maximize is the prior probability of the parameters that appears inside that integral $p\left(\mathbf{w}, \alpha, \sigma^2 | \mathbf{t}\right)$. It is not possible to apply Bayes rule to the last expression because the prior $p\left(\mathbf{t}\right)$ is unknown. Instead of trying this, expression 7.8 is used.

$$p\left(t_* | \mathbf{t}\right) = \int p\left(t_* | \mathbf{w}, \boldsymbol{\alpha}, \sigma^2\right) p\left(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2 | \mathbf{t}\right) d\mathbf{w} d\boldsymbol{\alpha} d\sigma^2 \tag{7.7}$$

$$p\left(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2 | \mathbf{t}\right) = p\left(\mathbf{w} | \mathbf{t}, \boldsymbol{\alpha}, \sigma^2\right) \cdot p\left(\boldsymbol{\alpha}, \sigma^2 | \mathbf{t}\right) \tag{7.8}$$

The first term in 7.8 can easily be computed since we have imposed $\mathbf{w}$ parameters distribution. Using the Bayes rule and conditional probability properties we find that $p\left(\mathbf{w} | \mathbf{t}, \boldsymbol{\alpha}, \sigma^2\right)$ distribution is Gaussian 7.9 with the parameters 7.11 and 7.10.

$$\begin{gathered} p\left(\mathbf{w} | \mathbf{t}, \boldsymbol{\alpha}, \sigma^2\right) = \frac{p\left(\mathbf{t}, \boldsymbol{\alpha}, \sigma^2 | \mathbf{w}\right) \cdot p\left(\mathbf{w}\right)}{p\left(\mathbf{t}, \boldsymbol{\alpha}, \sigma^2\right)} = \\ \frac{p\left(\mathbf{t} | \mathbf{w}, \boldsymbol{\alpha}, \sigma^2\right) \cdot p\left(\boldsymbol{\alpha}, \sigma^2 | \mathbf{w}\right) \cdot p\left(\mathbf{w}\right)}{p\left(\mathbf{t} | \boldsymbol{\alpha}, \sigma^2\right) \cdot p\left(\boldsymbol{\alpha}, \sigma^2\right)} = \frac{p\left(\mathbf{t} | \mathbf{w}, \boldsymbol{\alpha}, \sigma^2\right) \cdot p\left(\mathbf{w} | \boldsymbol{\alpha}\right)}{p\left(\mathbf{t} | \boldsymbol{\alpha}, \sigma^2\right)} = \\ \left(2\pi\right)^{\frac{-(N+1)}{2}} \cdot \left|\boldsymbol{\Sigma}\right|^{\frac{-1}{2}} \cdot \exp\left\{\frac{-1}{2}\left(\mathbf{w} - \boldsymbol{\mu}\right)^T \boldsymbol{\Sigma}^{-1}\left(\mathbf{w} - \boldsymbol{\mu}\right)\right\} \end{gathered} \tag{7.9}$$

$$\begin{gathered} \boldsymbol{\Sigma} = \left(\sigma^{-2}\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \mathbf{A}\right)^{-1} \\ \mathbf{A} = diag\left(\alpha_0, \alpha_1, \cdots, \alpha_N\right) \end{gathered} \tag{7.10}$$

$$\boldsymbol{\mu} = \sigma^{-2}\boldsymbol{\Sigma}\boldsymbol{\Phi}^T\mathbf{t} \tag{7.11}$$

Following with the other factor at 7.8, an approximation is needed to continue. We are going to assume that $p\left(\boldsymbol{\alpha}, \sigma^2 | \mathbf{t}\right)$ is very concentrated on the most probable values of $\boldsymbol{\alpha}$ and σ^2. In fact, we are really interested in 7.12.

$$p\left(t_* | \mathbf{t}\right) = \int p\left(t_* | \boldsymbol{\alpha}, \sigma^2\right) p\left(\boldsymbol{\alpha}, \sigma^2 | \mathbf{t}\right) d\boldsymbol{\alpha} d\sigma^2 \approx \int p\left(t_* | \boldsymbol{\alpha}, \sigma^2\right) \delta\left(\boldsymbol{\alpha}_{MP}, \sigma^2_{MP}\right) d\boldsymbol{\alpha} d\sigma^2 \tag{7.12}$$

In order to find the $\boldsymbol{\alpha}_{MP}$ and σ^2_{MP}, we are going to maximize the proportional expression at 7.13. Since the prior distributions of $\boldsymbol{\alpha}$ and σ^2 are uniform, only posterior distribution of $\mathbf{t}$ needs to be maximized 7.14.

$$p\left(\boldsymbol{\alpha}, \sigma^2 | t\right) \propto p\left(\mathbf{t} | \boldsymbol{\alpha}, \sigma^2\right) p\left(\boldsymbol{\alpha}\right) p\left(\sigma^2\right) \tag{7.13}$$

$$
\begin{aligned}
p\left(\mathbf{t}|\boldsymbol{\alpha},\sigma^2\right) &= \int p\left(\mathbf{t}|\mathbf{w},\sigma^2\right) p\left(\mathbf{w}|\boldsymbol{\alpha}\right) d\mathbf{w} \\
&= \left(2\pi\right)^{\frac{-N}{2}} \left|\sigma^2\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^T\right|^{\frac{-1}{2}} \exp\left\{-\frac{1}{2}\mathbf{t}^T \left(\sigma^2\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^T\right)^{-1} \mathbf{t}\right\}
\end{aligned}
\tag{7.14}
$$

7.4 Optimization

We need to maximize 7.14 over the hiper-parameters σ^2 and $\boldsymbol{\alpha}$. To do so, a numerical optimization method is needed to obtain the recursive updating equations 7.15 and 7.16. Note that μ_i makes reference to the i_{th} element of the vector 7.11 and Σ_{ii} to the i_{th} element of 7.10 diagonal.

$$\alpha_i^{new} = \frac{\gamma_i}{\mu_i^2} \tag{7.15}$$

$$\left(\sigma^2\right)^{new} = \frac{\left\|\mathbf{t} - \boldsymbol{\Phi}\boldsymbol{\mu}\right\|^2}{N - \Sigma_i\gamma_i} \tag{7.16}$$

$$\gamma_i = 1 - \alpha_i\Sigma_{ii} \tag{7.17}$$

In practice, updating 7.15 and 7.16 we generally find that many of the hiper-parameters α_i^{-1} turn numerically undistinguishable from zero. That means that the associated parameter w_i posterior distribution tends to be a delta centred at zero. That is the reason why we obtain a very sparse set of parameters $\mathbf{w}$, showing how only some samples of the training set are relevant, giving the name to the method Relevant Vector Machine or alternatively Sparse Bayesian Learning.

7.5 Making predictions and estimating the error

Now that we have obtained $\boldsymbol{\alpha}_{MP}$ and σ_{MP}^2, following from 7.7 we got 7.18. Both integrands are Gaussian, so $p\left(t_*|\mathbf{t},\boldsymbol{\alpha}_{MP},\sigma_{MP}^2\right) = \mathcal{N}\left(t_*|y_*,\sigma_*^2\right)$ with parameters 7.19 and 7.20. Taking in to account 7.18 and 7.19, $w_i = \mu_i$ are the parameters that make the best prediction of t_*.

$$p\left(t_*|\mathbf{t},\boldsymbol{\alpha}_{MP},\sigma_{MP}^2\right) = \int p\left(t_*|\mathbf{w},\sigma_{MP}^2\right) p\left(\mathbf{w}|\mathbf{t},\boldsymbol{\alpha}_{MP},\sigma_{MP}^2\right) d\mathbf{w} \tag{7.18}$$

$$y_* = \boldsymbol{\mu}^T\Phi\left(\mathbf{x}_*\right) \tag{7.19}$$

$$\sigma_*^2 = \sigma_{MP}^2 + \Phi\left(\mathbf{x}_*\right)^T \Sigma \Phi\left(\mathbf{x}_*\right) \tag{7.20}$$

As it was said at the first part of the chapter, one of the good properties of the RVM learning method is that it is a Bayesian method. That means that we have used the posterior probability properties to find the most probable solution to our learning problem. In addition, since we have computed since we have computed all posterior probabilities, we are able to estimate how good is our prediction with the variance 7.20. The error is a combination of the estimated error σ_{MP}^2 of the data and the uncertainty in the prediction $\sigma_{pred}^2 = \Phi\left(\mathbf{x}_*\right)^T \Sigma \Phi\left(\mathbf{x}_*\right)$.

Chapter 8

Results and conclusions

8.1 Algorithm summary and test design

To evaluate the performance of the method described in the last chapters, we are using a model learned with 40 images and we are testing it with with the others 57 images. For each image selected we will attempt to predict the gazed regions using the other selected images. In 8.1 some examples of the eye gazes on the x-ray images are shown. Next section will show the regression of the number of gazes on the watershed regions.

The first step is preprocessing all the images: aligning them to the chest model generated by the mean of all 99 chests marked on the images, doing the multi-resolution Sliding Band Filtering and finally post processing the output with the gazed points position probability density function. This probability density function has been estimated using the images out of the test set. Even though this is not a big change, is the fair option in order to estimate an unknown gazed.

Once preprocessed, features are extracted from the images. To do that, watershed segmentation is used to get the regions that will be used as samples. For each region in each image we extract the features presented in the past chapters.

Each region is treated as a sample to train the RVM model. Here we use the samples for the 40 images in the training set. And finally, the samples of the other images are used as a test to study the performance of the algorithm.

8.2 Results and error estimation

For each region of each image the error has been computed using the known target. Also, we know that predictive distribution of the RVM method is Gaussian (described before in this document) with mean the desired target and variance $\sigma^2 = \sigma^2_{MP} + \Phi\left(\mathbf{x}_*\right)^T \Sigma\Phi\left(\mathbf{x}_*\right)$. Knowing that $p(e < 2\sigma) = 0.9547$ for e zero-mean Gaussian with

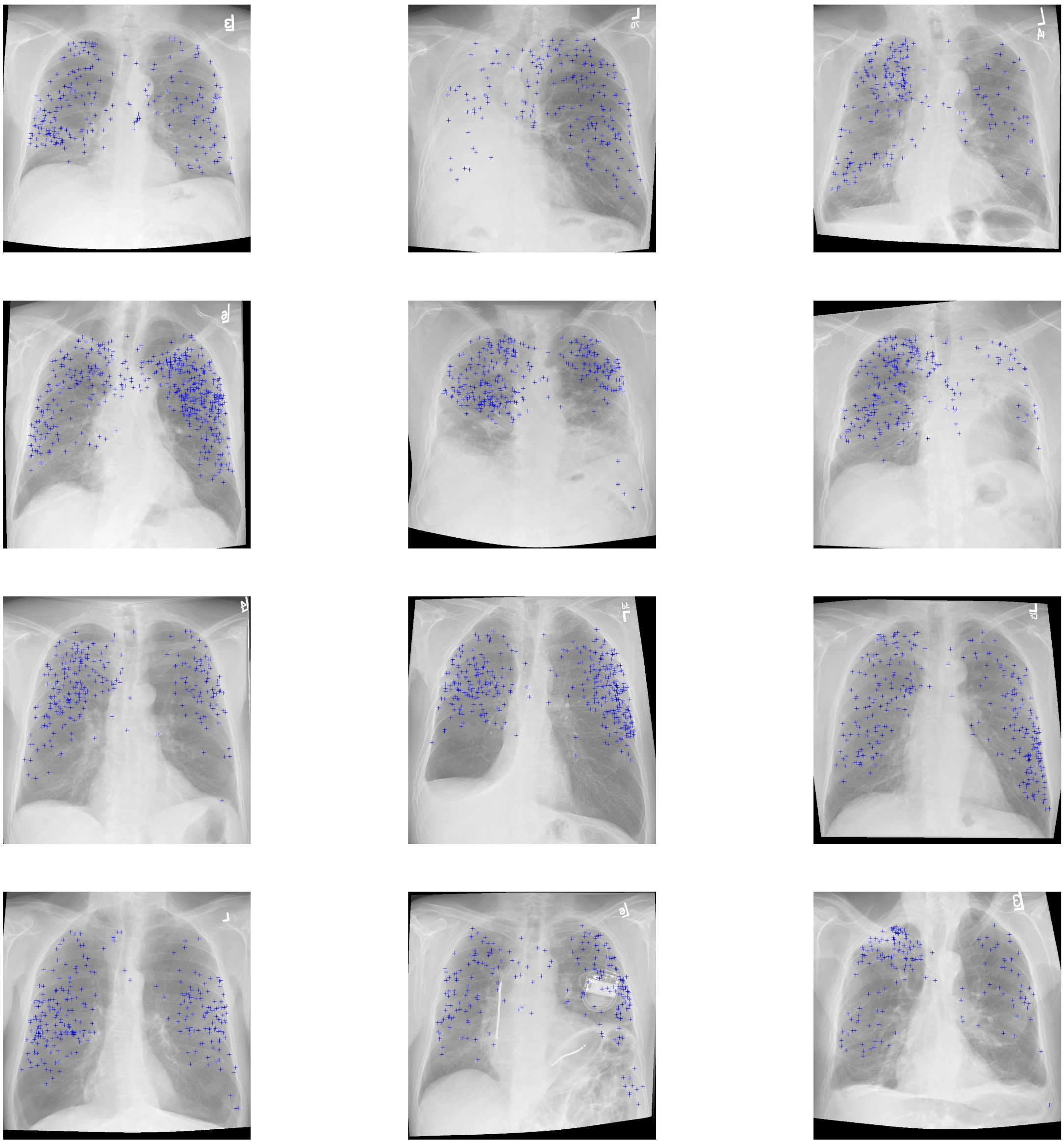

Figure 8.1: Gazed Points on aligned images

variance σ^2, we expect that the error $e = |y_* - t_*|$ will be lower than $2\sigma_*$ the 95.47% of the regions. Our experiment shows that the error committed is lower than 2σ in the 95.78% of the cases. That proves that our RVM model is highly reliable on expected error computing. In 8.2 it is shown how the error is lower than the error bar 2σ estimated for most of the samples.

The mean error committed by our algorithm 8.1 is 0.66 gazes per region. However,

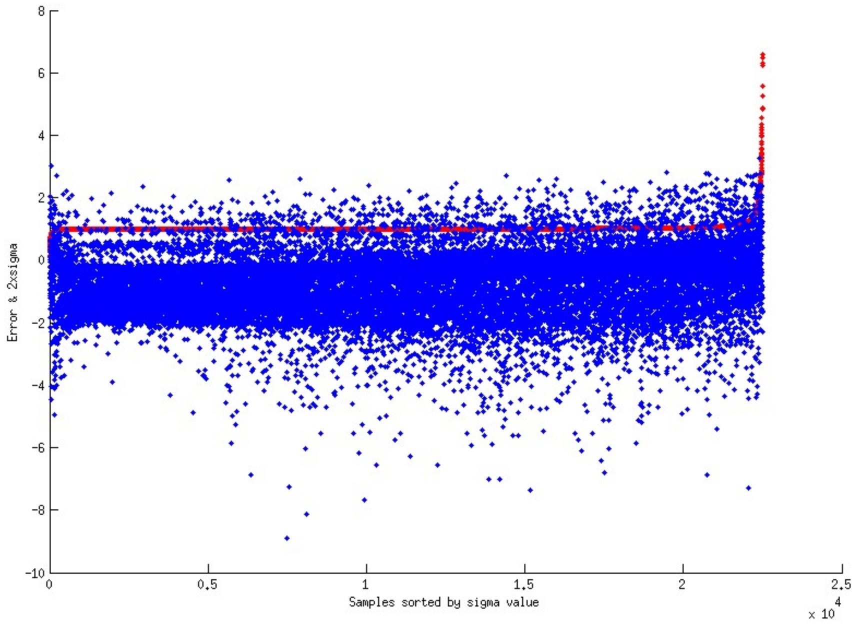

Figure 8.2: In blue: logarithm of the error 8.1 committed at each sample. In red: $log\,(2\sigma)$ estimated at each sample

the error is much bigger in the regions with more eye gazes. To evaluate that phenomena a special error measure has been used for the gazed regions (where there is one or more gazed pixels) 8.2. Computing the mean for all the gazed regions we found out a mean error of 47% per region. Also, the standard absolute error for the non gazed points is 0.54, bigger than computed with all the test set.

$$e = |y_* - t_*| \tag{8.1}$$

$$e^{gazed} = \frac{\left|y_*^{gazed} - t_*^{gazed}\right|}{t_*^{gazed}} \tag{8.2}$$

8.3 Conclusions

Analysing the performance of the algorithm, we find that for some cases it makes reasonably good predictions. The mean error of prediction is lower than 1 gaze per region (0.66 gazes), that is because our prediction is generally lower than 1 for the non gazed areas, which represent most of our samples are non gazed. Error rises in the gazed cases. Despite the fact that the absolute error is higher for the gazed regions, those samples get considerably higher predictions.

From a gazed-nongazed detector point of view, we can evaluate true positive and true negative rates. Considering a gazed prediction, the regions where the regression is higher than the mean value, we get a true positive rate of $65,61\%$ and a true negative rate of $62,91\%$. This suggest that our algorithm is able to show if a region is relevant or not.

Obviously the algorithm is still not useful for automatic diagnosis or for diagnosis assistance, but those are promising results that show that the convergence index filters are a good approach to cancer and abnormal regions detection on a chest x-ray images. Also, the multi-resolution algorithm has given us a good result identifying objects with different sizes. Both methods are key steps to get that results. As an example of how the algorithm works on the images shown in 8.1, we can see the regression output and the target (number of gazes in the region) in 8.3, 8.4, 8.5 and 8.6. The gray level in the images corresponds to the number of gazes scaled to 255, so $Im_{out} = 255 \cdot \frac{Im_{in}}{max\{Im_{in}\}}$. That way, we see what relevance (in gazes number) has each region compared with the others seen on the image.

8.3.1 Future work

Despite the regression results are disappointing, the preprocessing algorithm developed has obtained quite promising results. Based on that preliminary treatment of the images, some work could be done to get more accurate regression values. This work could be resumed in:

1. Use of a more specific data base with lung cancer cases. Also, it could be interesting to use the exact location of the cancer instead of the eye gaze of the doctors.

2. Redefining the problem: Multi-resolution Sliding Band Filters have shown good results in nodule highlighting. Also, RVM has been proved to be useful to discriminate relevant regions in the image. However, the number of gazes per region regression is quite inaccurate. Redefining the target could be the answer to improve regression results. It is reasonable to think that the number of gazes in a watershed region is highly correlated to the features of the other regions in the same image. A clear disease could take a lot of attention from other zones that would became more interesting for the doctors in other contexts.

3. Try segmentation alternatives: Even though watershed seem to work quite well, in some cases the output regions do not correspond to a possible nodule. That is because we are forced to average the SBF output in order to eliminate the noise

to get watershed regions reasonably big. After filtering, the watershed regions do not represent possible nodules in flat areas where there is nothing else but noise. So region size, perimeter and eccentricity are not related with the gazed points in those cases. To solve that, a different kind of segmentation could be developed based on the points of maximal gradient convergence used during the Sliding Band Filtering process.

4. As a long term objective, automatic landmark finding should be developed to automatize the alignment process.

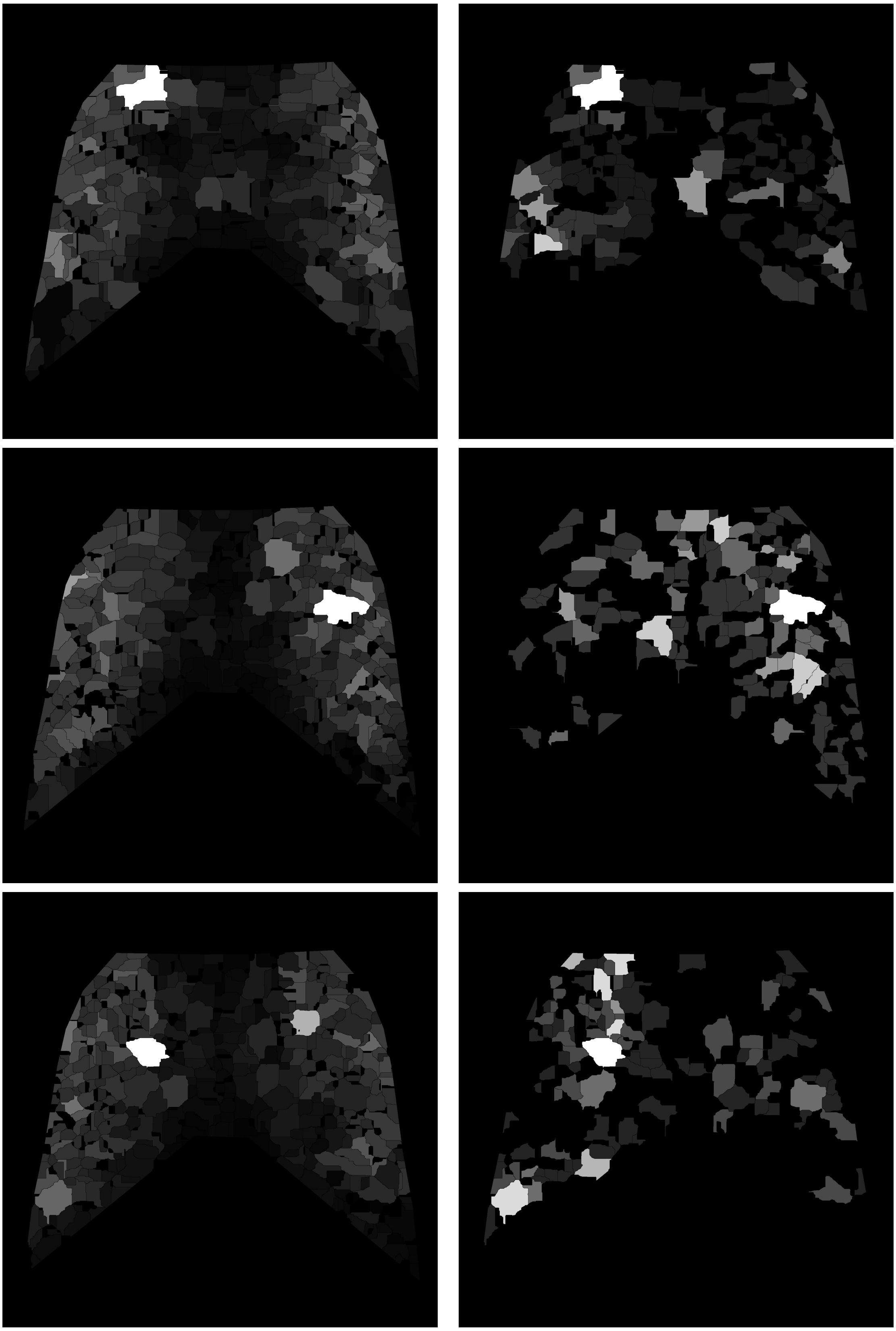

Figure 8.3: Regression (left column) and target (right column) for a 3 chest x-ray images

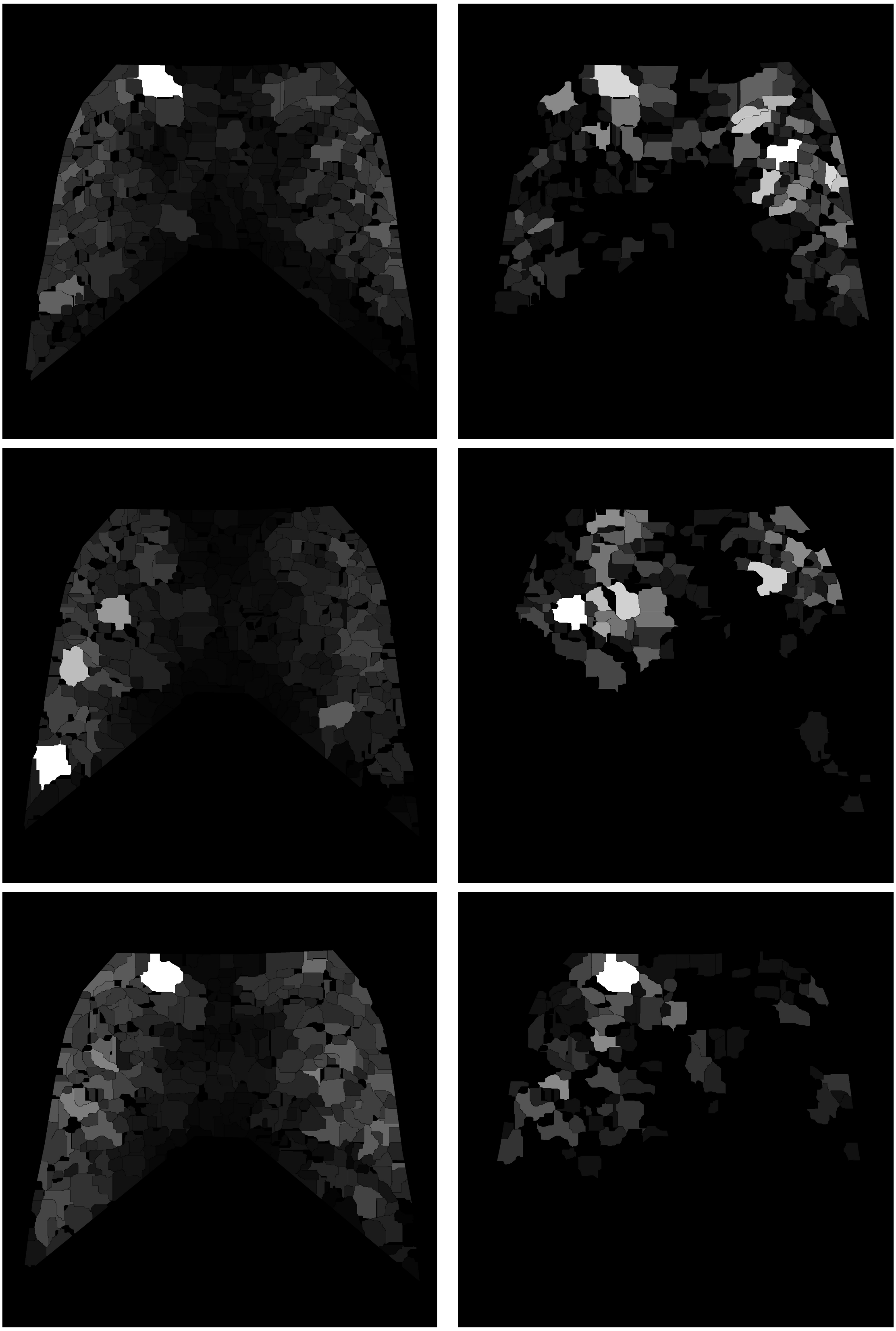

Figure 8.4: Regression (left column) and target (right column) for a 3 chest x-ray images

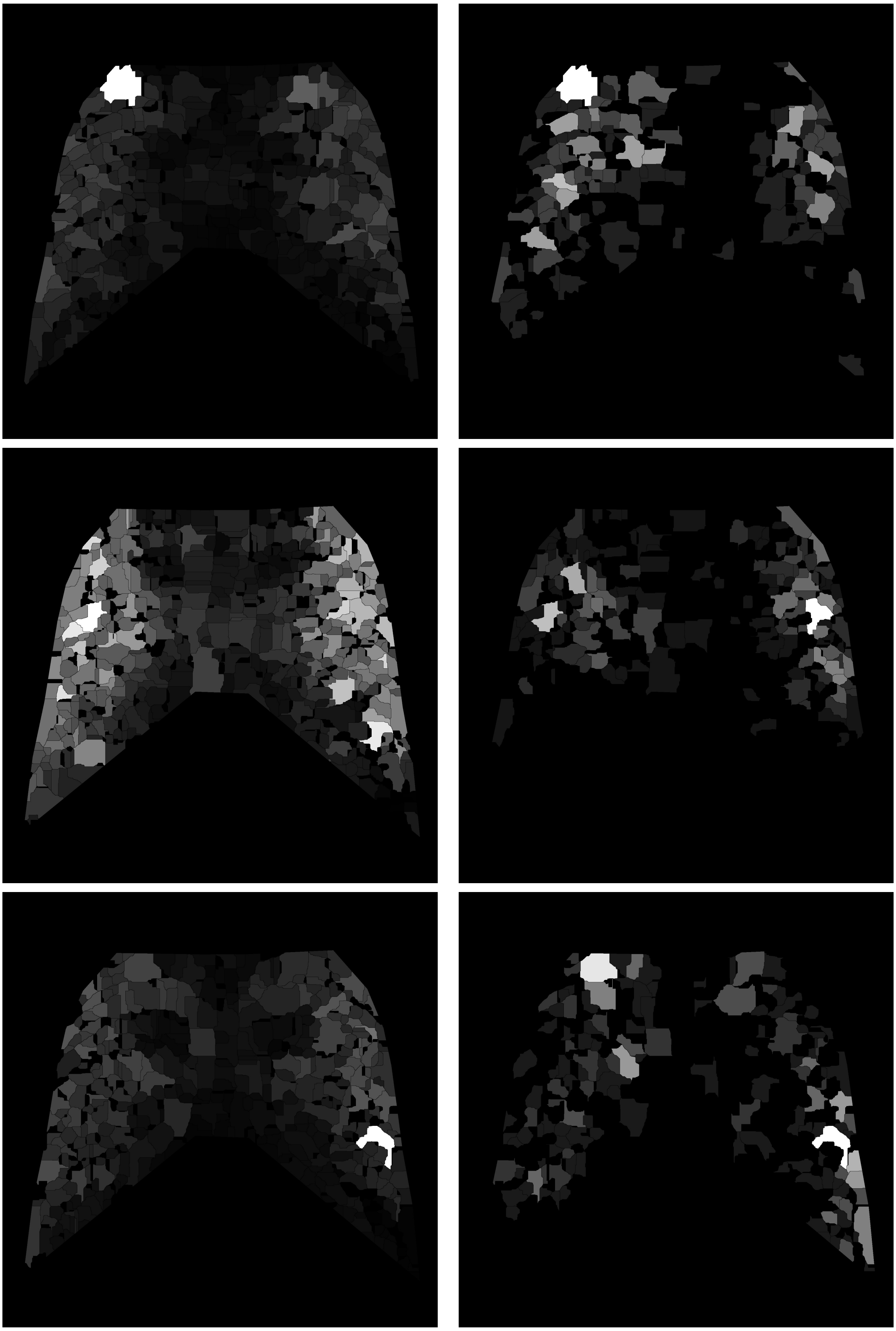

Figure 8.5: Regression (left column) and target (right column) for a 3 chest x-ray images

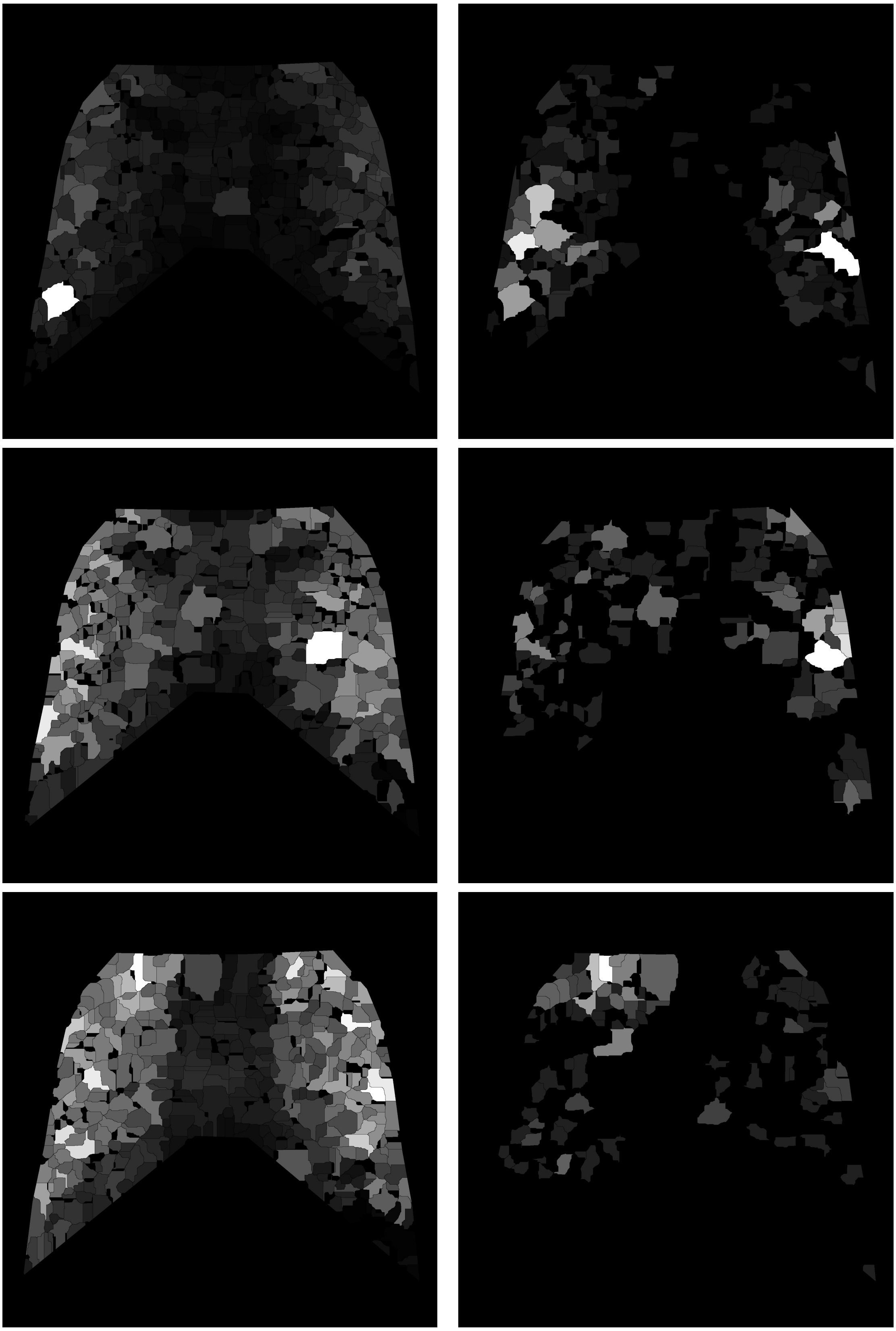

Figure 8.6: Regression (left column) and target (right column) for a 3 chest x-ray images

Bibliography

[1] Nonlinear adaptive convergence index filters and their characteristics. In *Proceedings of the International Conference on Pattern Recognition - Volume 3*, ICPR '00, pages 3526–, Washington, DC, USA, 2000. IEEE Computer Society.

[2] Jun Wei 0002, Yoshihiro Hagihara, and Hidefumi Kobatake. Detection of cancerous tumors on chest x-ray images candidate detection filter and its evaluation. In *ICIP (3)*, pages 397–401, 1999.

[3] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. Wiley Interscience, 2 edition, 2000.

[4] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition, 2001.

[5] Anil K. Jain. *Fundamentals of digital image processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1989.

[6] H. Kobatake and S. Hashimoto. Convergence index filter for vector fields. *Trans. Img. Proc.*, 8(8):1029–1038, August 1999.

[7] H. Kobatake and M. Murakami. Adaptive filter to detect rounded convex regions: Iris filter. In *Proceedings of the 13th International Conference on Pattern Recognition - Volume 2*, ICPR '96, pages 340–, Washington, DC, USA, 1996. IEEE Computer Society.

[8] Carlos S. Pereira, Hugo Fernandes, Ana Maria Mendonça, and Aurélio Campilho. Detection of lung nodule candidates in chest radiographs. In *Proceedings of the 3rd Iberian conference on Pattern Recognition and Image Analysis, Part II*, IbPRIA '07, pages 170–177, Berlin, Heidelberg, 2007. Springer-Verlag.

[9] kanupriya Raturi, K. Susmitha Wils, and Abhilash Mt. Detection of lung nodules using image enhancement. In *Proceedings of the 8th conference on Signal processing, computational geometry and artificial vision*, ISCGAV'08, pages 183–186, Stevens Point, Wisconsin, USA, 2008. World Scientific and Engineering Academy and Society (WSEAS).

[10] Michael E. Tipping. Sparse bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.*, 1:211–244, September 2001.

[11] Jun Wei, Yoshihiro Hagihara, and Hidefumi Kobatake. Detection of rounded opacities on chest radiographs using convergence index filter. In *Proceedings of the 10th International Conference on Image Analysis and Processing*, ICIAP '99, pages 757–, Washington, DC, USA, 1999. IEEE Computer Society.