Computer Vision
Report on Incisor Segmentation Project

Laavanya Wudali Lakshmi Narsu (r0690809)
Master in Artificial Intelligence

June 2018

Contents

1 Introduction

1.1 Description of the problem

In this project, the main task is to extract the incisors from dental radiographs. A dental panoramic radiograph with a labelled incisor is shown in Fig.1. Thus the segmentation algorithm must output a binary image that labels the pixels inside the incisors.
During the course we have seen several segmentation methods largely based on clustering methods. Point distribution based models are one way of doing model based segmentation and are suitable for this task. Active shape models are deformable point distribution models which are built from annotated images [1]. These models can then be used on unseen images by iteratively deforming around strong edges in the image to narrow down to the target pattern.

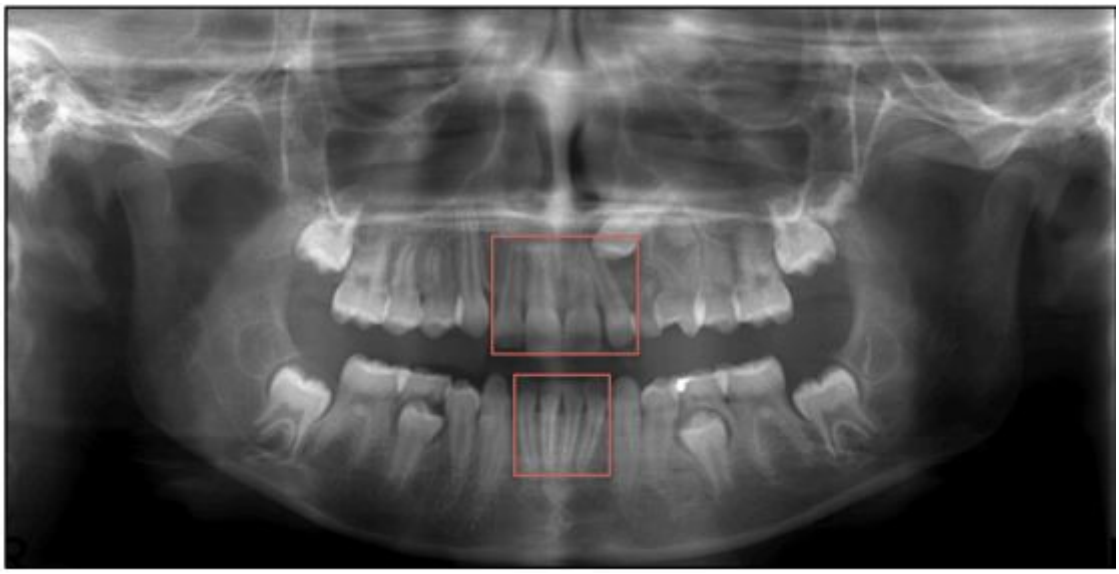


Figure 1: Dental panoramic radiograph with the labeled incisors

1.2 Data

In this subsection we are presenting the data that was available for the project. We had three different types of data, the landmarks, the radiographs, and the segmentation of the training images.

1. Landmarks: The landmarks are provided as <x, y>list of coordinates. For each tooth there are a total of 40 landmarks.
2. Radiographs: These are the original images used for both training and testing. The images are approximately about 1600 * 3000.
3. Segmentations: These are binary images with the non-black part being the segmented tooth and are used to evaluate the model.

2 Methodology

2.1 Point Distribution Model

Principal Component Analysis is a statistical dimensionality reduction technique. Given a $n \times d$ matrix of n observations and d features, it transforms this matrix into n observations of $k < n$ variables that capture most of the variance of the data.
In order to achieve this, PCA performs an eigendecomposition of the covariance matrix and retains the eigenvectors corresponding to the largest eigenvalues. These eigenvectors are orthogonal as the matrix is symmetric, and form basis vectors for the transformed feature space. The higher an eigenvalue it is, the more the variance that is captured by the corresponding basis eigenvector.
In the context of shape models, we can use PCA to limit the variability of the model by transforming the shape from the space of its landmarks to the space the selected eigenvectors [1]. In this space, the shape represented by a vector of dimensionality $2l$,(l is the number of landmarks), can be represented as in terms of the mean shape and a linear combination of the selected eigenvectors P based on the weight vector b of the size of the number of prinicipal components.

$$x = \bar{x} + Pb \tag{1}$$

b can be varied to represent deviations from the mean. In order to not allow too much deformation of the model, each element in this vector is limited by the corresponding eigenvalue of the principal component. As the square root of the eigenvalue represents the standard deviation of the data, each component $|b_i| < 3\sqrt{\lambda}$.

2.2 Procrustes Alignment

Before the point distribution model can be constructed from given training set, the landmark shapes corresponding to different images that are in different coordinate systems, need to be *aligned* or converted to a common coordinate space by appropriate linear similarity transformations [1]. This is called Procrustes Analysis. A shape $X_i = [(x_1, y_1)...(x_l, y_l)]$ can be aligned to a shape $X_j = [(x'_1, y'_1)...(x'_l, y'_l)]$ by finding a transformation composed of a rotation, scaling and a translation.

$$T(x) = \begin{pmatrix} scos\theta & -ssin\theta \\ ssin\theta & scos\theta \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} = \begin{pmatrix} a & -b \\ b & a \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \tag{2}$$

that minimizes the squared error $||T(x_i) - x_j||_2$. This can be done by solving the following set of linear equations [2].

2.3 Fitting

The fitting algorithm is an iterative algorithm that is based on two fitting procedures, and can be described as follows:

1. Fit Grey Model.
2. Fit Shape Model, updating the transformation matrix (translation, rotating, and scaling) and b.

3. Constraint b parameter to plausible values as explained in PCA section.

Iterate until convergence, or certain iteration limit. This algorithm is applied for every point in our shape model [1].

In order to fit the grey model we look along the normal of each point. This normal is computed taking into account the boundary points, in our case 3 points on each side. After computing the normal, we look into the grey-level values along the normal, and find the best-fit possible. This summarizes the fitting the grey model.

The second step, the shape model, is transforming our previous shape updating the parameters stated before in the "fit shape model" step. This is performed as per equations X, Y found in the alignment section of this project. It is important to make sure as stated in point 3 of the algorithm, that the values of b remain plausible ($< 3\sqrt{\lambda}$). If an element of b exceeds this value, is placed back to the maximum possible, which as said is $3\sqrt{\lambda}$.

2.4 Preprocessing of Dental Radiographs and image transformations for Grey level profiles

In order to compute the derivative along the normal we use the sobel derivative. Then the derivative is scaled by the absolute values in order to account for intensity differences. The derivative must have a peak at the correct location owing to the transition from black to white(tooth). As the derivatives are noisy even after the application of the smoothed derivative, we applied a median filter with size 5, to accentuate regions with high values of derivatives, and suppress small regions.

2.5 Initialization

The solutions of the ASM are very sensible to the initialization methods used. We used two categories of initialization.

- **Manual**: Using the test landmarks to generate the initialization. This served the purpose to check whether the algorithm was working properly, but the results are highly biased by giving the correct landmarks as initialization instead of looking for an initial position based on grey level, or using the mean shapes.
- **Automatic template matching**.We used an initializing method, without having in consideration the test landmarks. We use the mean image of all the training data to build a template. In order to fix the template size we used the coordinates of the extremes of our training landmarks and resized the mean image to fit the size of the bounding box. This template was then slid across each test image and the point of highest normalized cross correlation were found.The mean training shape was then translated to the center of the template location as found by the template match. We generated two types of initializations using this technique. The first one, generating the bounding box for all eight teeth, meaning both top and bottom teeth. The other one, tries to be more accurate with the bounding box, and generates two bounding boxes, one for the upper teeth, and another one for the bottom teeth.

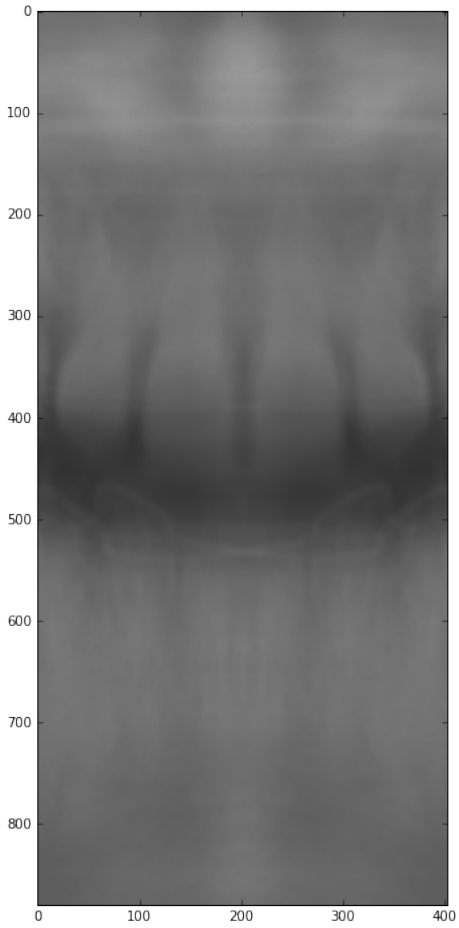


Figure 2: Bounding Box eight teeth

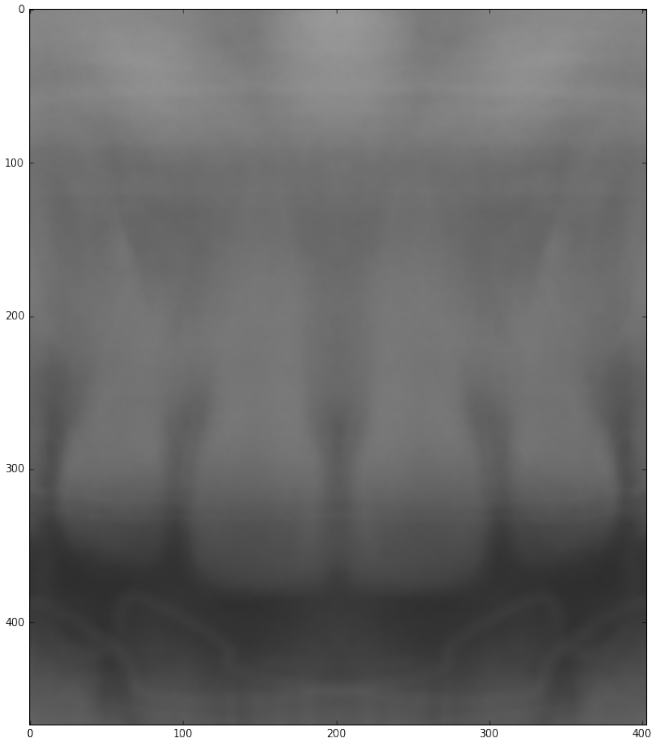
In the next section we will explore all the work done, and present all the relevant figures that explain the work done.

3 Evaluation and Results

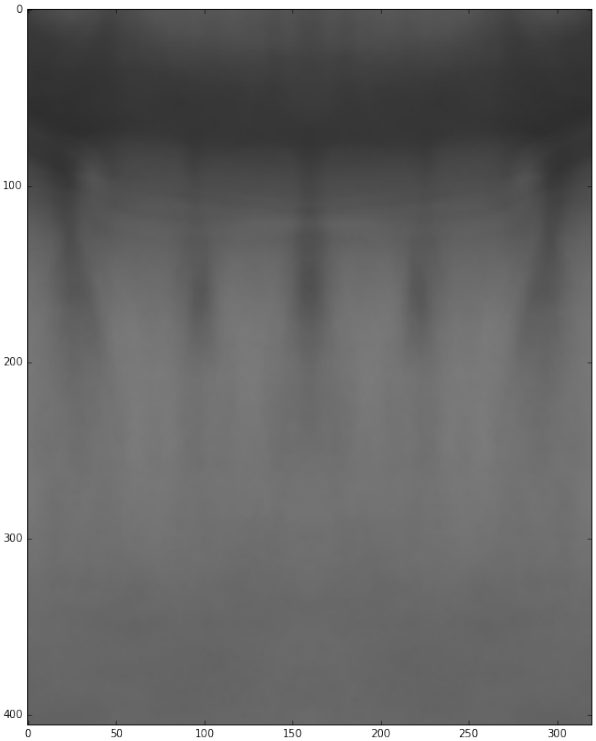
In this section all the intermediate results will be shown. With that, we will show all the analysis done, such as PCA, or grey Level analysis and then finally, the main results.

3.1 Initialization

The first subsection will present the bounding box (BB) method of initialization. Figures 2 3 present the Bounding Box analysis, Fig. 2 being bounding boxes of the eight teeth, and Fig. 3 being the two split bounding boxes.

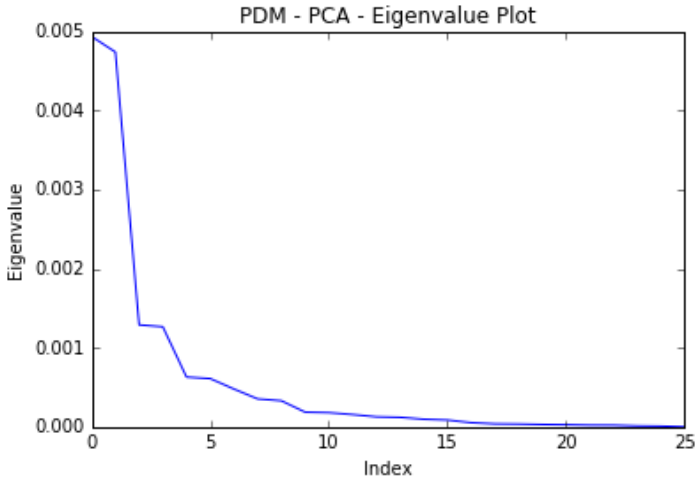


(a) Top Bounding Box

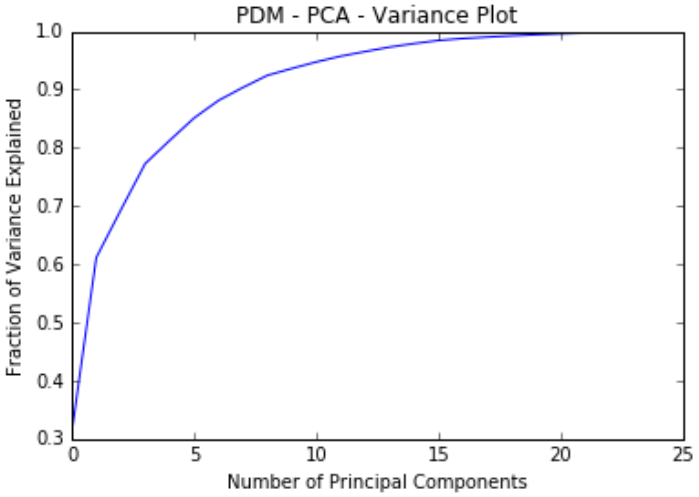


(b) Bottom Bounding Box

Figure 3: Initializing with 2 Bounding Boxes



(a) Eigenvalue results



(b) Eigenvalue results

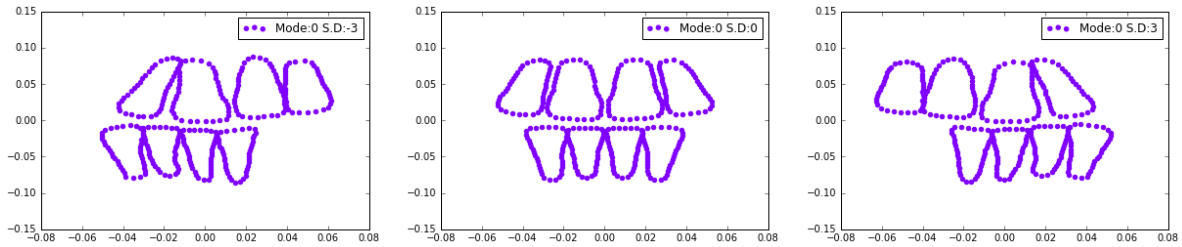
Figure 4: Results of PCA

3.2 PCA and Shape Analysis

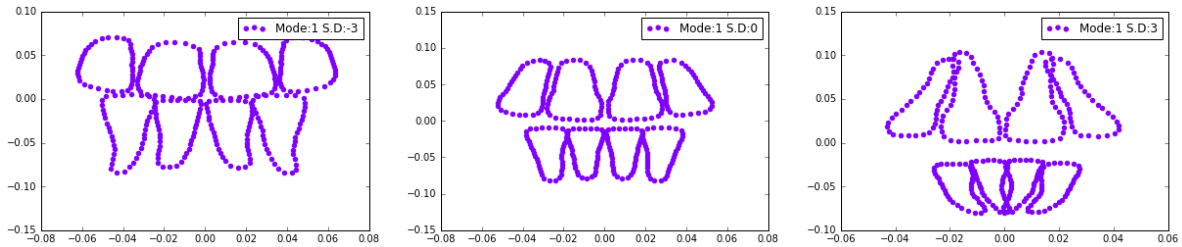
Figure 4 presents the results of PCA done to the landmarks. The original matrix shape is $30 * 640$. 15 images were given and with them all the pairs of landmarks, but we were also given the mirrored landmarks, hence $15 * 2 = 30$. The 640 comes from 320 pairs of x, y coordinates. Figure 4 shows that we capture capture nearly all the variance of the dataset with 15 components, and that we capture almost 60% of the variance with just three components.

Figure 5 shows what happens when we work with parameter b in the following equation $x = \bar{x} + Pb$. Each row represents what happens when we modify b, the first row, affecting the first PC, the second row, the second PC and the third row, the third PC. We show the extreme cases with $b = \pm 3\sqrt{\lambda}$.

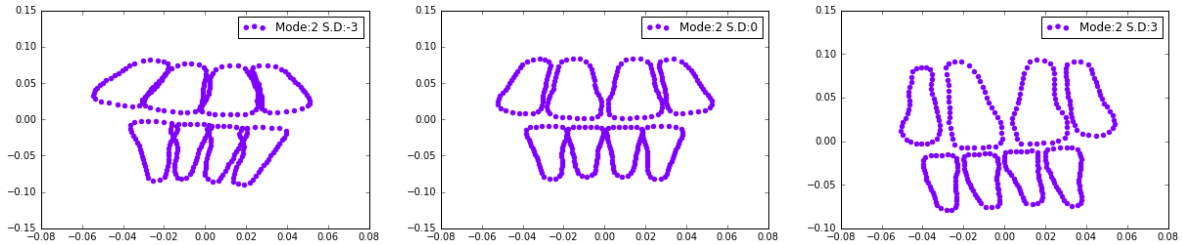
- First component affects the alignment of the teeth, they are moved left or right



(a) First PC, $b = -3\sqrt{\lambda_1}$ (b) Mean values (c) First PC, $b = +3\sqrt{\lambda_1}$



(d) Second PC, $b = -3\sqrt{\lambda_2}$ (e) Mean values (f) Second PC, $b = +3\sqrt{\lambda_2}$



(g) Third PC, $b = -3\sqrt{\lambda_3}$ (h) Mean values (i) Third PC, $b = +3\sqrt{\lambda_3}$

Figure 5: Shape Analysis

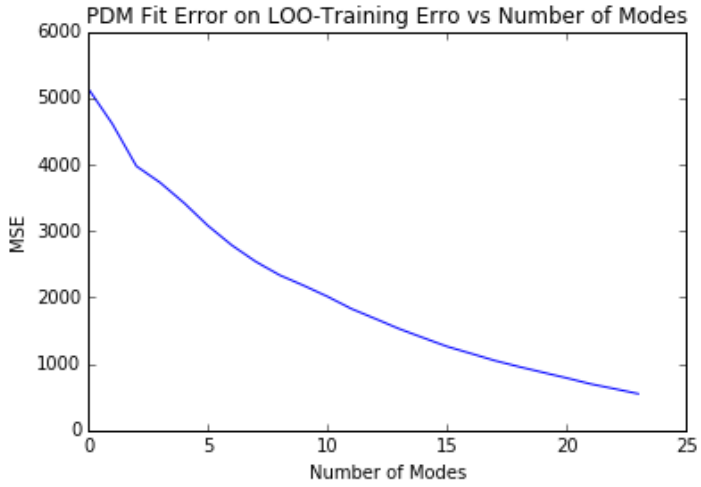
depending on b.

- Second component affects multiple things, like the height of the teeth considering the center the mean value, figures b, e and h.
- The third component works on the direction of having the teeth being more separated the bigger b parameter is.

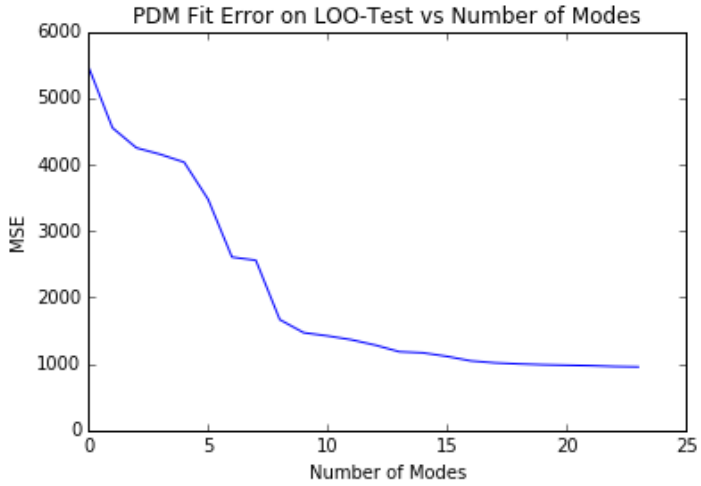
3.3 Leave-one-out Fit Analysis

This section will provide an analysis of the accuracy results achieved in the train and test, based on the number of Principal Components used with the model.

Figure 6 shows what was expected, the more modes (principal components) that we use the better the results. It is noticeable that in the test set, after 16 principal components, our results seem to saturate, meaning that the addition of extra PC do not add significat information. This is consistent with the fact that with 16 PC we capture more than 99% of the variance of the dataset. This setup is the one we use for the rest of the analysis. The values in the y axis represent the root squared mean error between the original landmarks



(a) Train Accuracy



(b) Test Accuracy

Figure 6: Point Distribution Model Accuracy based on PC

and the fit.

3.4 Evaluation

In order to separate the data into training images and test images, we used the cross-validation method, more specifically, Leave-One-Out. So all images except one, went for training, and the remaining image was used for testing purposes. This way we made sure that the training data is not used on the testing part, and it gave us enough iterations to have an estimate of our accuracy models.

For the evaluation purposes we implemented the Jaccard index, or Jaccard similarity coefficient.

$$J(A, B) = |A \cap B|/|A \cup B| \tag{3}$$

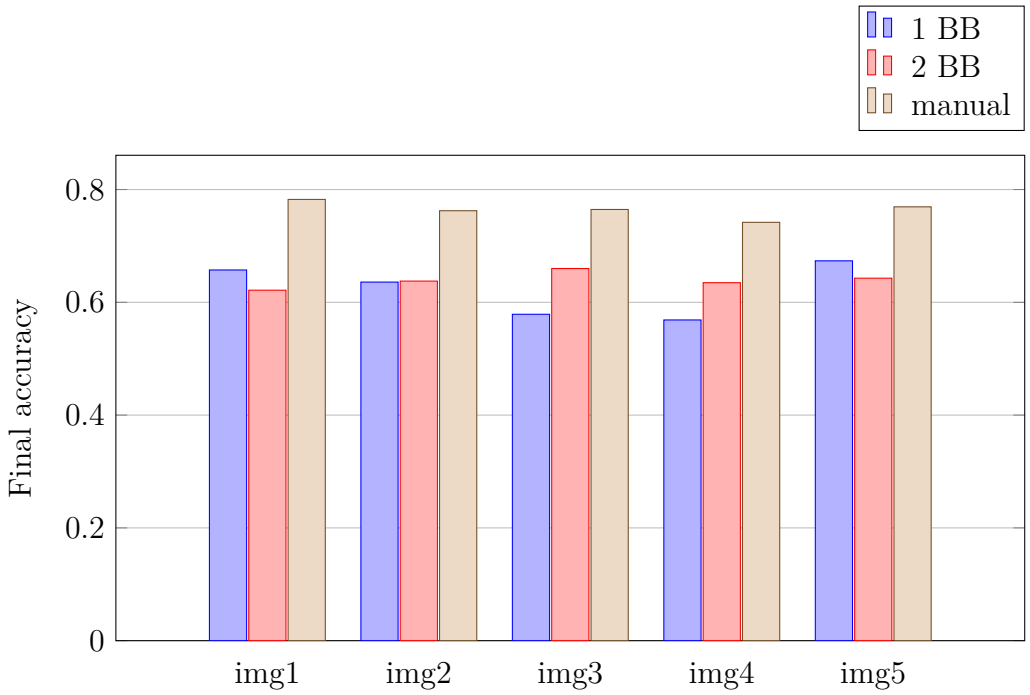


Figure 7: Final Accuracy according to different initialization methods

This index is used to compute the similarity of two sets, and it is closely related to the Dice coefficient presented in the ASM paper [1].

In this following section we will be able to observe the final results obtained from the fitting process. The first set of graphs represents the final accuracy. This is seen in 7 8 and 9

In the next set of figures, represented in 10 and 11, we can see the final models after the algorithm has finished.

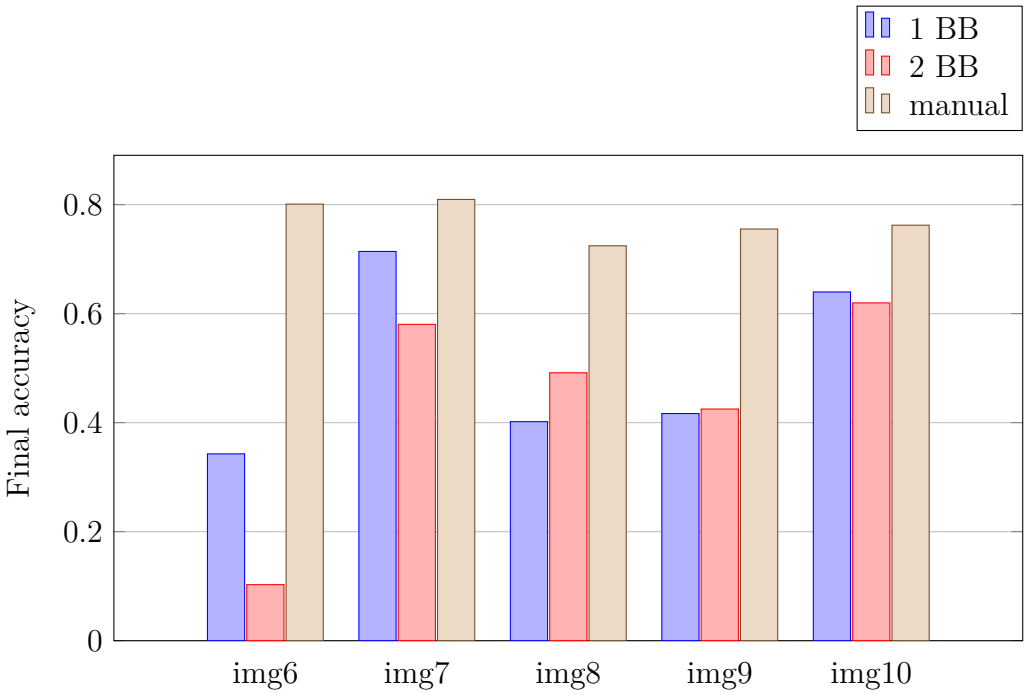


Figure 8: Final accuracy according to different initialization methods

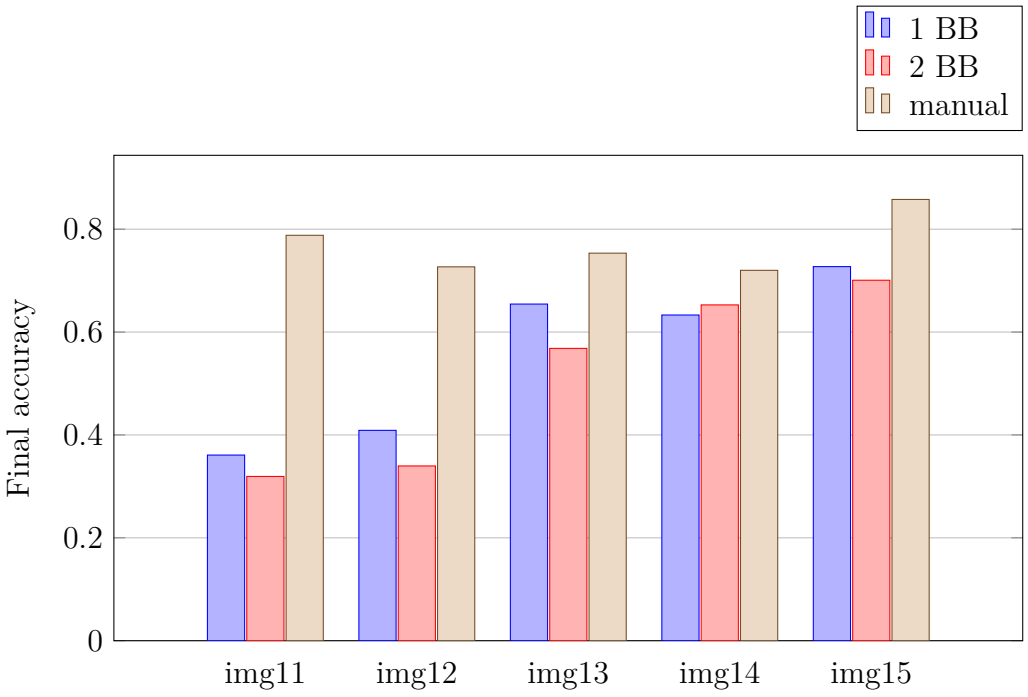
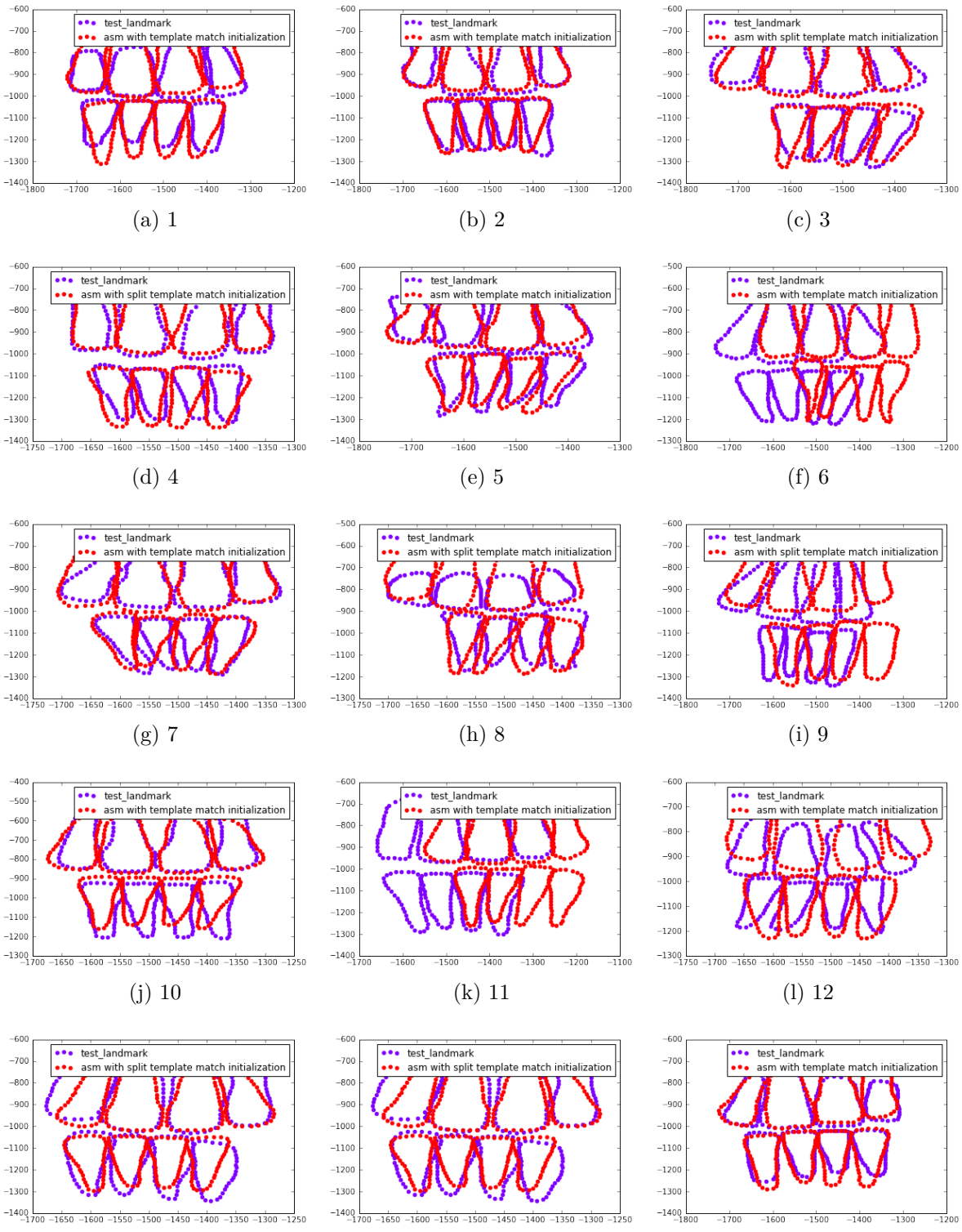


Figure 9: Final accuracy according to different initialization methods



(a) 1 (b) 2 (c) 3

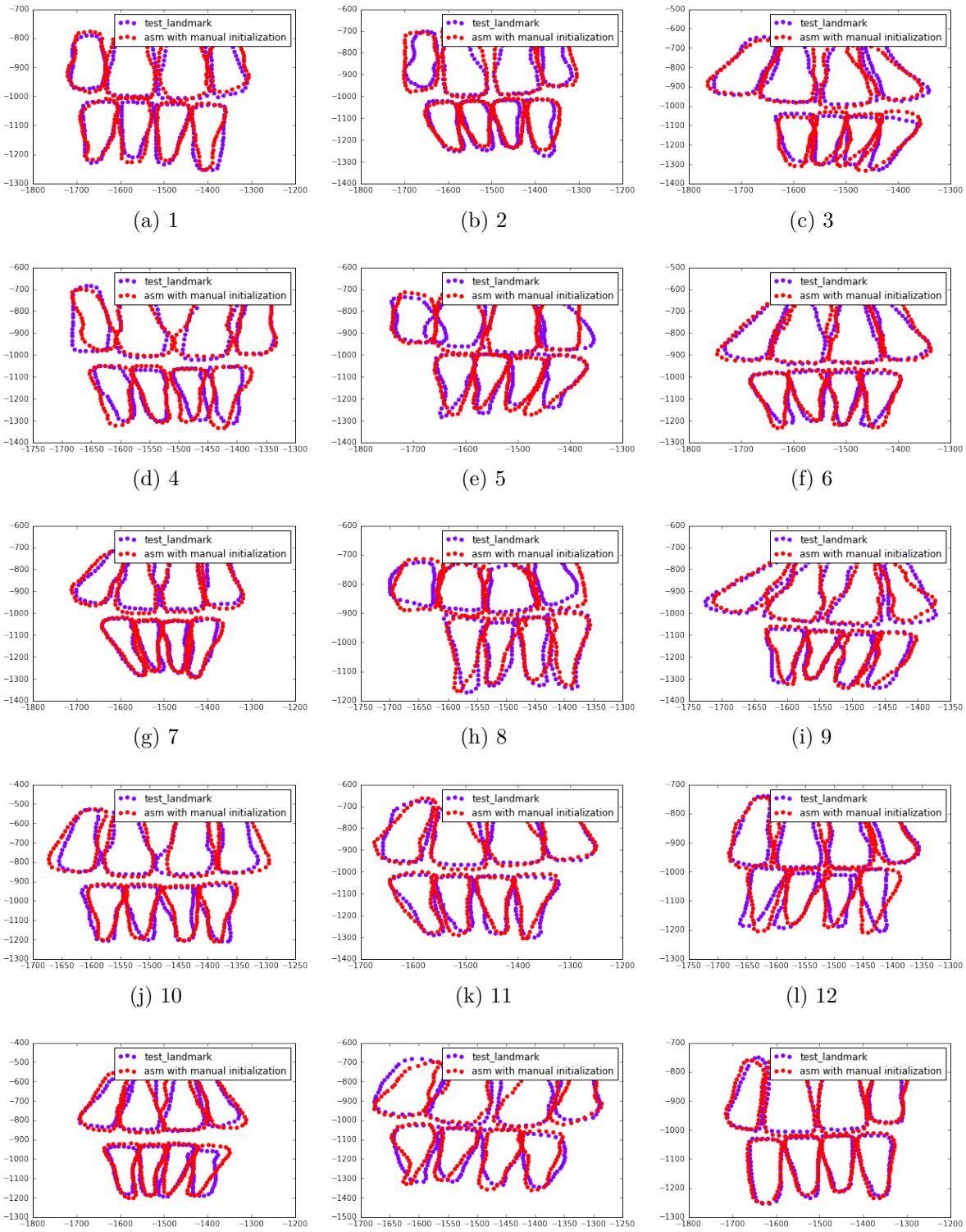
(d) 4 (e) 5 (f) 6

(g) 7 (h) 8 (i) 9

(j) 10 (k) 11 (l) 12

(m) 13 (n) 14 (o) 15

Figure 10: Final results using Bounding Boxes



(a) 1 (b) 2 (c) 3

(d) 4 (e) 5 (f) 6

(g) 7 (h) 8 (i) 9

(j) 10 (k) 11 (l) 12

(m) 13 (n) 14 (o) 15

Figure 11: Final results with manual initialization

4 Conclusions

As we have seen during the presentation several conclusions can be dragged of. The most important of them, is that the most important part of the process, is the initialization. If the model is initialized very close to the optimal solution, the algorithm works fine, otherwise, it is easy to observe that can present some different shapes than the ones expected. Our results have proved to be different when initializing close, or on top of the expected shape, and when initializing not as close as needed.

This leads to a second point, where we can also say that the automatic initialization is not trivial, specially in medical images. Just by looking at the pictures one can see, that there is no fixed center, and that each image is completely different, and even with a small dataset, only a small percentage of images are close to our mean incisor model.

Furthermore tuning the parameters for the initialization turned out really hard. With big numbers we gave too much freedom to the teeth, and with small numbers, teeth were not moving too much.

About PCA, we found hard to select a proper number of components, since the more components we added, the less error we were having and we couldn't fine any breaking point on which to stop selecting components.

References

[1] T. Cootes, E. Baldock, and J. Graham. An introduction to active shape models. *Image processing and analysis*, pages 223–248, 2000.

[2] T. F. Cootes, C. J. Taylor, et al. Statistical models of appearance for computer vision, 2004.