



Katholieke  
Universiteit  
Leuven

Department of  
Computer Science

# INCISOR SEGMENTATION

Computer Vision : Project (H02K5a)

**Kevin Boets (r0298464)**  
**Tim Lesage (r0294518)**

Academic year 2014–2015

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Alignment</b>	<b>2</b>
<b>3</b>	<b>Convergence</b>	<b>2</b>
<b>4</b>	<b>Initializing starting position</b>	<b>2</b>
<b>5</b>	<b>Contour fitting</b>	<b>2</b>
5.1	Nearest edge based . . . . .	3
5.2	Energy based . . . . .	3
5.2.1	Gradient size based . . . . .	3
5.2.2	Orientated gradient size based . . . . .	3
5.3	Derivative based . . . . .	3
<b>6</b>	<b>Results</b>	<b>3</b>
<b>7</b>	<b>Conclusion</b>	<b>3</b>
<b>8</b>	<b>References</b>	<b>3</b>

# 1 Introduction

We needed to design an algorithm that is capable of automatically segmenting the 8 incisors on a radiograph. To accomplish this we were provided with 14 radiographs with the landmarks for each incisor. We used these landmarks to make an active shape model, section 2. Before using these models for fitting the incisors of other radiographs, we made a method to find an initial guess for each incisor, section 4. When we have found the initial position, we then use the active shape models to fit to the incisors on the image, section 5. We tried different fitting algorithms, multiple variations of an energy based algorithm, section 5.2 and one derivative based algorithm, section 5.3. When implementing these algorithms we encountered some problems with convergence, section 3. After some experimentation we decided that the derivative based algorithm gave the best results and thus we used this algorithm to do some more extensive tests, section 6.

## 2 Alignment

To be able to combine all the different landmarks of each incisor they need to be comparable (same center, similar scale and rotation). We realigned the landmarks so they have the same center, rotation and scale. To accomplish this we used Procrustes Analysis as described in protocol 4 of [4]. Convergence in step 7 of this protocol is achieved if every element of the new result differs at most 0.01 from the result of the previous iteration.

## 3 Convergence

When testing the different algorithms, we encountered a problem with the fitting of our contours: it didn't converge completely. To solve this we first were a bit lenient about when convergence happened: if less than 10% of the points describing the contour changed, we considered this converged. This approach gave us some better results, but it didn't solve the problem completely. We thought that the contours endlessly cycled between certain states. We solved this by introducing an extra condition: if each point of the contour changed at most 10 pixels in every 50 iterations, we considered this converged. This new condition guaranteed termination of the algorithm.

A second observation we made was that the contour would move outside the bounds of a reasonable solution. When this occurs, we would stop the execution of the fitting and consider the current result as best fit. We consider the bounds of a reasonable solution as a rectangle with three times the width and 140% the height of the initial rectangle found as initial position, see section 4.

## 4 Initializing starting position

Before we can try to fit our model to the image, we need to find an initial estimate of the incisors. We decided to use PCA to find the estimate. To make sure teeth aren't confused with each other we decided to group the upper and lower incisors in two different groups and search for each group independently. This means that we extract two cropped images, respectively of the four upper and lower incisors. In figures ?? and ?? we show the results of both PCA procedures.

To limit the amount of comparisons to be done and attain reasonable solutions, we can limit the search space to the central area of each image, since the incisors only reside there. The areas for both upper and lower incisors are displayed in figure ??.

To extract an estimate for each tooth, we simply divide the found result in four equally long parts, figure ??. Even though it is not precise, we decided that it would be good enough as an initial estimate.

## 5 Contour fitting

After finding the initial estimate of an incisor, we then have to fit the estimate to image. We tried four different approaches to this particular problem. The first one is based on using the nearest edges, second and third one are based on an energy function and the fourth one is based on derivative. Although they are slightly different, they still all follow the same algorithm: the active shape model algorithm described in protocol 2 from [4].

## 5.1 Nearest edge based

The first approach was to iteratively examine the area of each model point and calculate the distance to the nearest edge. We then moved each model point to its best neighbor with the lowest distance. Then we mapped the model point generated by the pca to our found points as described in [3]. The edges were found by using the canny operator provided by opencv [1]. To reduce noise we have tested a variety of filters: GaussianBlur, MedianBlur and bilateralFilter[2]. A problem that arose was that the incisor had an internal edge around the crown ?? and that the edges around the roots were weak. We tried to solve this by processing the root and crown differently but to no avail.

## 5.2 Energy based

### 5.2.1 Gradient size based

### 5.2.2 Orientated gradient size based

## 5.3 Derivative based

# 6 Results

# 7 Conclusion

# 8 References

## References

- [1] [http://docs.opencv.org/modules/imgproc/doc/feature\\_detection.html?highlight=canny#canny](http://docs.opencv.org/modules/imgproc/doc/feature_detection.html?highlight=canny#canny).
- [2] <http://docs.opencv.org/modules/imgproc/doc/filtering.html>.
- [3] Computer vision [h02k5a] : Final project: Model reconstruction.
- [4] Tim Cootes. Model-based methods in analysis of biomedical images. In Oxford University Press, editor, *Image Processing and Analysis*. Oxford University Press.