Katholieke
Universiteit
Leuven

**Department of
Computer Science**

INCISOR SEGMENTATION

Computer Vision : Project (H02K5a)

Kevin Boets (r0298464)
Tim Lesage (r0294518)

Academic year 2014–2015

Contents

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 a 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 profile based algorithm, section 5.3. When implementing these algorithms we encountered some problems with convergence, section 3. After some experimentation we decided that the energy based algorithm using the oriented gradient, 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 the same center, rotation and scale. To accomplish this we used Procrustes Analysis as described in protocol 4 of [5]. 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 convergence 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. We choose 10 pixels over 50 iterations, because we wanted to only continue when a significant change has happened during a considerable amount of iterations.

A second observation we made was that the contour would move outside the bounds of a reasonable solution, figure 1. 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 trice the width and 140% the height of the initial rectangle found as initial position, see section 4. We chose such a big difference in width and height, because some incisors are rotated, which could drastically affect the width.

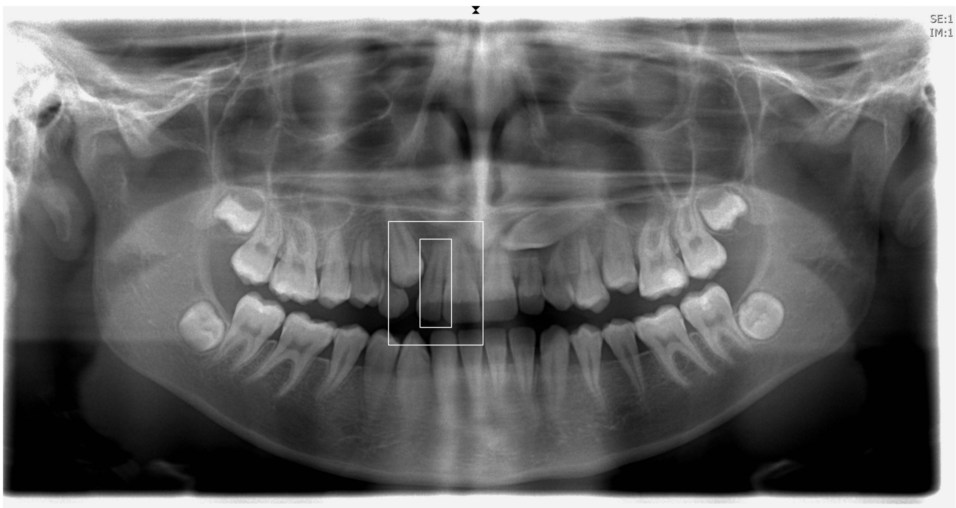


Figure 1: The estimate of an incisor and the reasonable boundaries

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.

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 2.

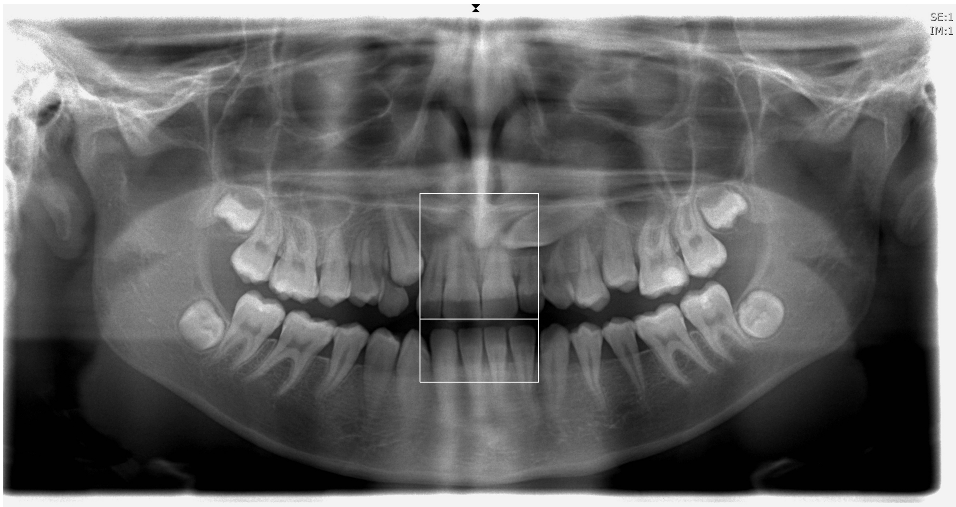


Figure 2: The search space for the upper and lower incisors.

To extract an estimate for each tooth, we simply divide the found result in four equally long parts, figure 3. Even though it is not precise, we decided that it would be good enough as an initial estimate.

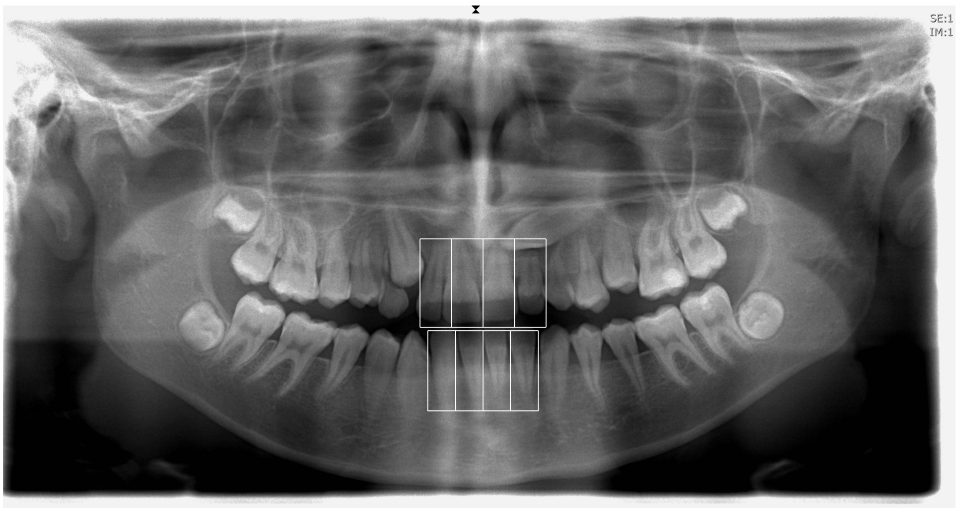


Figure 3: Example of how we divide the found initial position.

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 profiles. Although they are slightly different, they still all follow the same algorithm: the active shape model algorithm described in protocol 2 from [5].

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 it's best neighbor with the lowest distance. Then we mapped the model generated by the PCA to our found points as described in [3]. This was done until the contour converged. 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, see figure 4, 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.

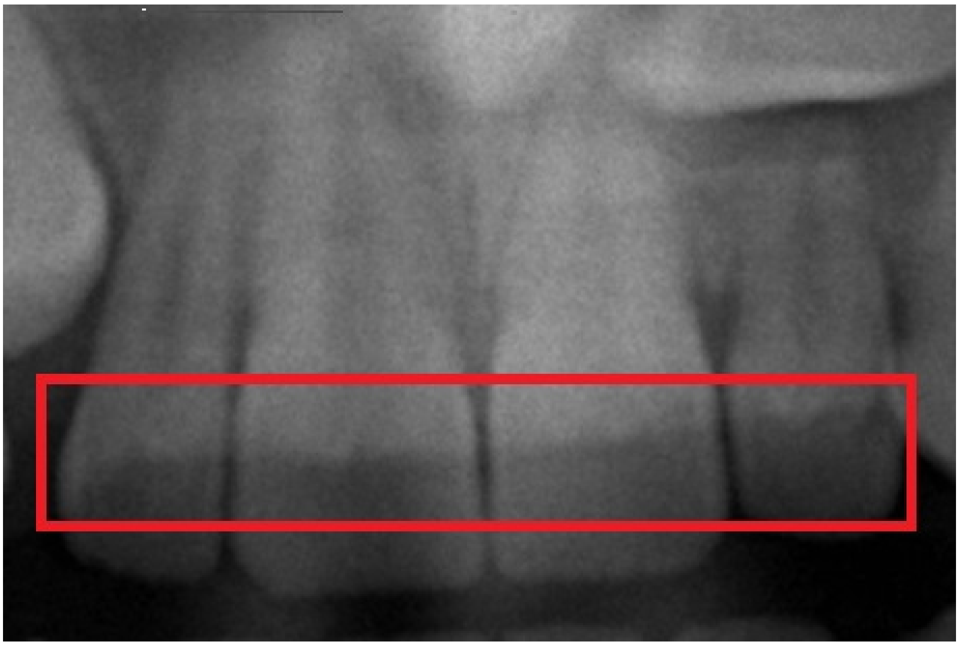


Figure 4: The edges around the crown.

5.2 Energy based

The second and third approaches are energy based. We examine the area around each point and search for the point that will result in the lowest total energy for the entire contour. To efficiently find the lowest, we used the Viterbi algorithm. After finding the nearest contour with the lowest energy, we map the model generated PCA to the found contour. This is repeated until the contour converges.

The energy is divided in two parts: internal and external energy. The internal energy is defined by the formula used in the course, stiffness and elasticity. For the external energy we tried two different functions, which is the only difference between the two approaches.

To acquire the gradients we used the sobel function from [2]. This function already does some filtering before calculating the gradients. To avoid to much filtering, we decided not to add any extra filters.

5.2.1 Gradient size based

Our first energy based approach uses the same external energy function as seen in the course:

$$E_{external}(v) = -(|G_x(v)|^2 + |G_y(v)|^2) \tag{1}$$

This function is only based on the magnitude of the gradients.

5.2.2 Orientated gradient size based

Our second energy based approach uses an external function we found in [4].

$$E_{external}(v) = -\|G(v)\|cos(\theta_G(v) - \theta_N(v)) \tag{2}$$

In this equation we use $\theta_G(v)$, the gradient direction at v, and $\theta_N(v)$, the contour's normal direction at v. These direction will decide in which direction the point will be pushed. When $\theta_G(v)$ is in the same direction as $\theta_N(v)$ the point will be pushed in the direction normal, if not it will be pushed in the opposite. We hoped to benefit by using the direction of the gradient, which should always be toward the exterior of an incisor. It would eliminate the possibility of using the edges of neighboring teeth.

5.3 Profile based

For our last approach we use the profile method from [5]. To improve the position of points, we look along profiles normal to the boundary of the contour. To lose the effect of global intensity changes we take the derivative along the profile and normalize it. We sample profiles from the training set and build a statistical structure, mean($\overline{g}$) and covariance(S_g), for each point. This structure will be used to find the most similar position along the profile. The similarity is based on the quality of the fit:

$$f(g_s) = (g_s - \overline{g})S_g^{-1}(g_s - \overline{g}) \tag{3}$$

The profiles we take from the training set contains 5 pixels in each direction along the normal, which results in a profile of length 11. When we try to find the best position we take a profile at the current point containing 10 pixels in each direction, profile length 21. We then sample parts of length 11 from this profile and from these samples we find the most similar one. The center pixel of the most similar sample will be picked as the new point.

6 Results

We weren't sure which of these three approaches was the best: nearest edge, oriented gradient and profiles. To learn which one was the best, we decided to do a leave-one-out analysis for all three approaches. The results of this analysis would give us the necessary information. We calculated the 2-norm of the distance between our found model points and the real landmarks. We put the results in table 1.

$$Error = \|landmarks - foundpoints\| \tag{4}$$

We decided beforehand that the one with the least error would be selected. Table 1 shows that the energy based approach with the orientated gradient gives the best results. Even though it is better than the others, it is still not very good. After looking through the images generated during the analysis, we saw a few mismatches between the incisors and our initial positioning. We assume that these mismatches are the main reason for the big error differences between the different radiographs. We display this in figures 5 and 6.

To show the end results of our work we compare the segmentations we find with the given landmarks for each radiograph, see figure 7 and 8.

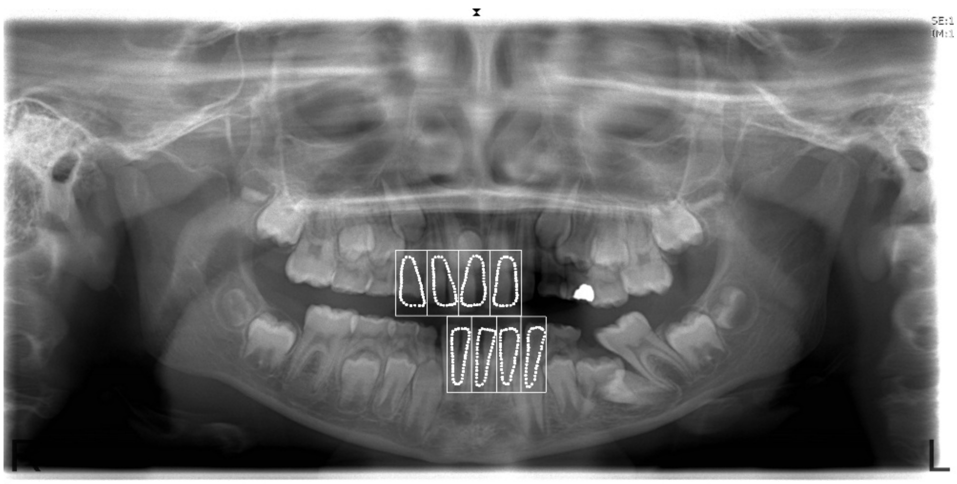


Figure 5: The upper incisors are matched to much to the left.

Radiograph	Orientated Gradient	Nearest Edge	Profiles
1	1819.655258	1901.343545	2355.863966
2	2574.327681	2516.855612	3339.389387
3	3881.933749	4144.124856	3174.753026
4	4176.542958	4363.965605	4777.121959
5	1896.22025	2614.866113	2893.772849
6	4110.183924	4037.398879	4307.558375
7	3245.977556	3732.113309	3738.139635
8	2385.99068	3049.260544	3168.804846
9	2449.894409	3457.718605	3211.818036
10	3459.987205	3998.65734	4211.293344
11	3034.125283	3532.33652	3647.065266
12	1657.372988	2346.401805	1971.689986
13	4290.749684	4288.406321	4682.828212
14	3844.508912	3226.977932	2987.349513
Som	42827.47054	47210.42699	48467.4484
Gem	3059.105038	3372.173356	3461.9606

Table 1: Leave-one-out analasys, Error calculated with equation 4

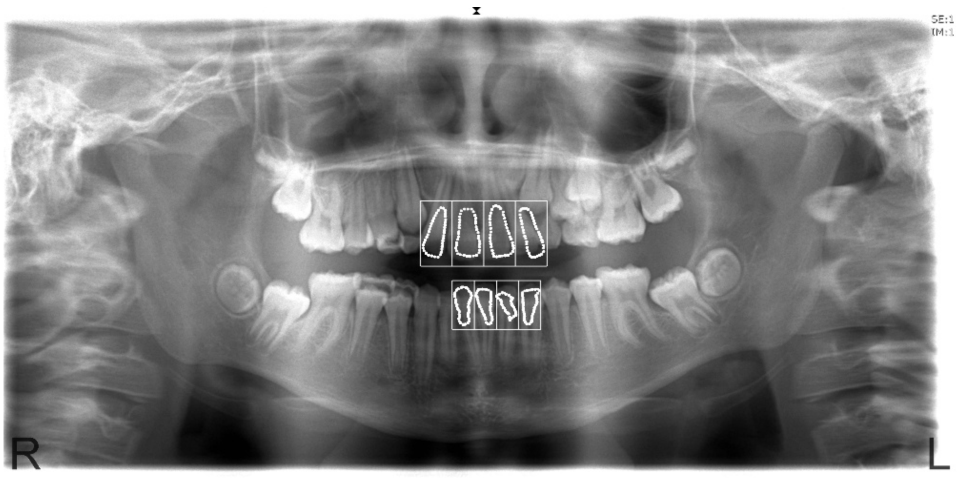
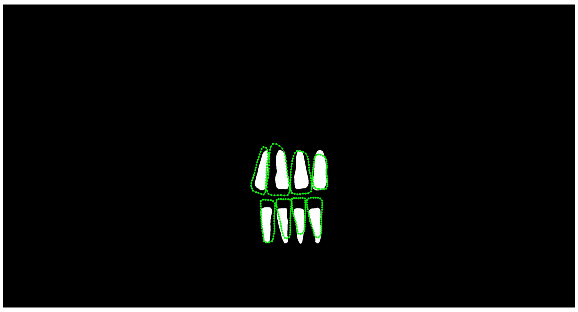


Figure 6: The lower incisors are to short.

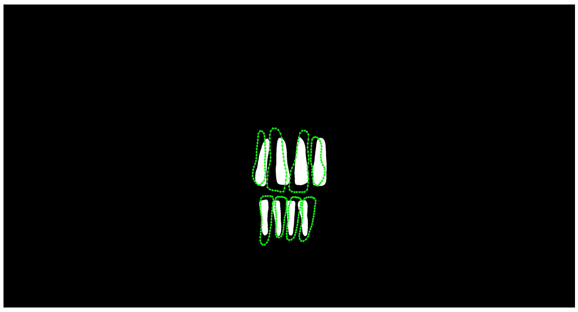
7 Conclusion

We can conclude that our algorithm needs some improvement and refinement. We definitely need to improve the automatic initialization of the incisors. Although the contour finding needs some improvement, we think it can be limited by modifying the weights of the energy function. Unfortunately due to time constraints we couldn't pursue this anymore.

Figure 7: The comparison between our found incisors (white) and the real incisors (green) for:

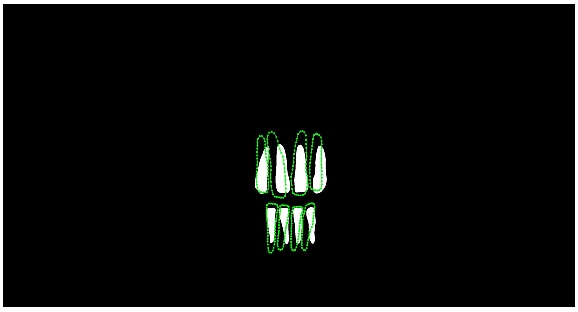


(a) radiograph 1.

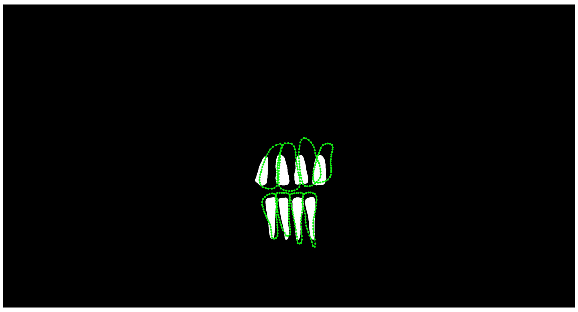


(b) radiograph 2.

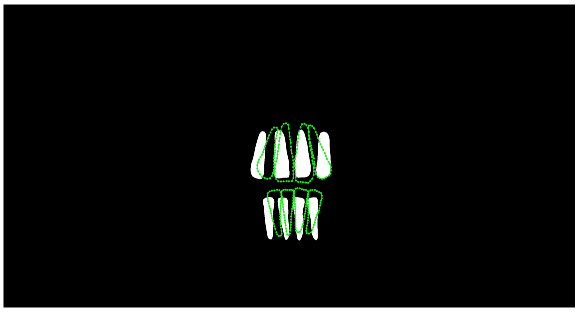
(c) radiograph 3.



(d) radiograph 4.

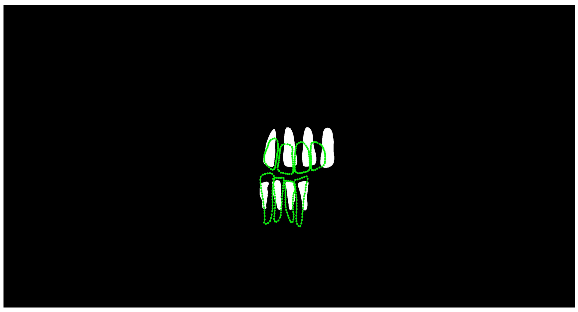


(e) radiograph 5.



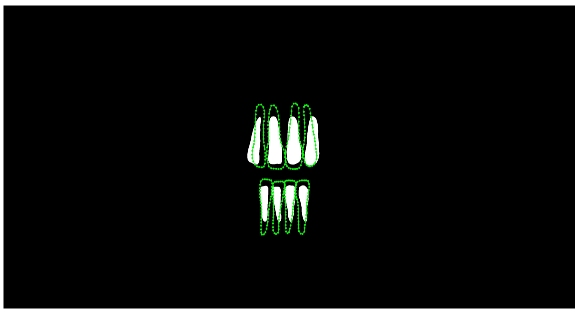
(f) radiograph 6.

(g) radiograph 7.



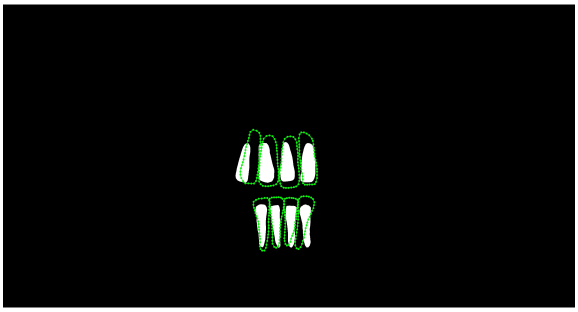
(h) radiograph 8.

(i) radiograph 9.

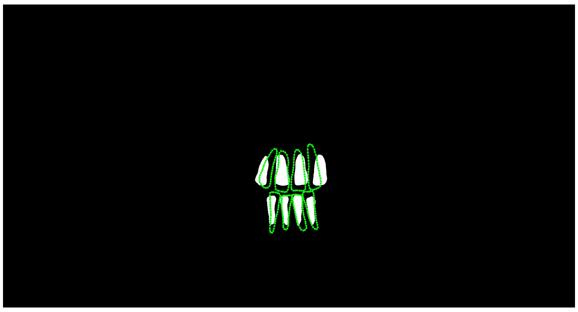


(j) radiograph 10.

Figure 8: The comparison between our found incisors (white) and the real incisors (green) for:

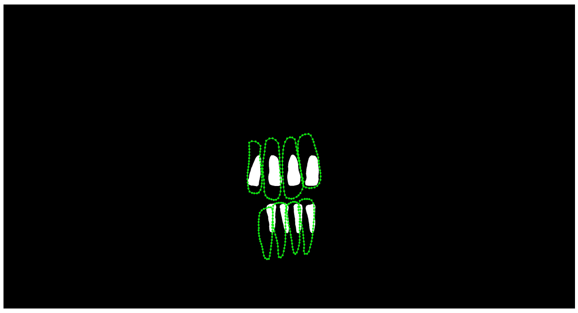


(a) radiograph 11.



(b) radiograph 12.

(c) radiograph 13.



(d) radiograph 14.

8 References

References

[1] `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] Hong Chen and Anil K. Jain. Tooth contour extraction for matching dental radiographs. `http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1334581&tag=1`.

[5] Tim Cootes. Model-based methods in analysis of biomedical images. In Oxford University Press, editor, *Image Processing and Analysis*. Oxford University Press.