Automatic Cephalometric X-Ray Landmark Detection Challenge 2014: A machine learning tree-based approach

Rémy Vandaele[1,2], Raphaël Marée[1], Sébastien Jodogne[2], and Pierre Geurts[1]

[1] Systems and Modeling, Department of EE and CS & GIGA-R,
[2] Department of Medical Physics, C.H.U.,
University of Liège,
Sart-Tilman, B34
4000 Liège, Belgium
remy.vandaele@ulg.ac.be

Abstract. In this paper, we describe the machine learning approach we used in the context of the Automatic Cephalometric X-Ray Landmark Detection Challenge. Our solution is based on the use of ensembles of Extremely Randomized Trees combined with simple pixel-based multi-resolution features. By carefully tuning method parameters with cross-validation, our approach could reach detection rates $\geq 90\%$ at an accuracy of 2.5mm for 8 landmarks. Our experiments show however a high variability between the different landmarks, with some landmarks detected at a much lower rate than others.

1 Introduction

Cephalometry is a particular process to analyze the human cranium. It consists in detecting landmarks and measuring the distances (or distance ratios) between these landmarks in order to detect possible problems or to plan intervention treatments. This technique is heavily used in the medical domain. Typically, the detection of the landmarks is done manually, which makes it a very complicated and time-consuming task. There is therefore a strong interest to develop automated or semi-automated landmark detection methods. While several approaches have been proposed in the literature to solve this problem, there is no systematic comparison between all these methods, which makes the choice of a suitable detection algorithm difficult.

In this context, the goal of the present challenge is to propose a fair assessement of landmark detection algorithms on a gold standard evaluation dataset. Challenge participants are provided with a dataset of 100 manually annotated 1935×2400 pixels images to train their models and are asked to provide landmark predictions for 100 test images before the challenge workshop and for 100 additional test images during the workshop.

The problem of landmark localization in X-Rays has been extensively studied in the literature. Existing methods are typically based on the combination

of template matching algorithms and prior knowledge information and differ mainly in the feature extraction methods (see [5] for a brief review). In contrast, our solution is based on the application of generic machine learning methods, in particular tree-based ensemble methods (e.g., Random Forests [1] or Extremely Randomized Trees [4]). Randomized decision forests have found many applications in computer vision, mainly because of their flexibility, robustness to irrevelant features, low computational complexity and high expressive power [2]. They have been already successfully used for interest point or landmark detections, e.g in [8] to perform morphometric measurements in microscopy images of Zebrafish or in [3] to predict organ location in three dimensional CT images.

We describe our algorithmic solution in Section 2 and then present in Section 3 the protocol we adopted to generate our final model for the challenge. Section 4 reports some cross-validation results.

2 Method

Following our previous work [8], we adopted a supervised learning approach that exploits the manually annotated images to train models able to predict landmark positions in new, unseen, images. In particular, a separate pixel classification model is trained for each landmark to predict whether a given image pixel corresponds to the position of this landmark or not. This model is trained from a learning sample composed of pixels extracted either in the close neighboorhood of the landmark or at other randomly chosen positions within the training images. Each pixel in the training sample is described by a vector of visual features at different resolutions.

The different steps of the algorithm for a single landmark are explained in the following subsections. This procedure is repeated for every landmark separately.

2.1 Extraction and description of pixels

Each observation in the training sample corresponds to a pixel at position (x, y) in one of the training images and is labeled into one class among $\{0, 1\}$ and described by several input features. We described below successively the output class associated to each pixel, the input features used to describe them, and the pixel sampling mechanism.

Output classes. In principle, only one position in each image corresponds to the landmark, which means that if N training images are available, only N positive examples will be available to train our pixel classification model. To extend the set of positive examples, we consider as positive examples all pixels that are at a distance at most R from the landmark, where R is a method parameter. More precisely, if the landmark is at position (x_l, y_l) in an image, then the output class of a pixel at position (x, y) in the same image will be 1 if $(x-x_l)^2+(y-y_l)^2 \leq R^2$, 0 otherwise.

Input features. A pixel at location (x, y) will be described by raw pixel values in a square subwindow of height and width $2W+1$ centered at the shifted position $(x + t_x, y + t_y)$, where W, t_x, and t_y are method parameters. Because of the introduction of the shift parameters t_x and t_y (that will be tuned by cross-validation), the model is potentially able to detect the landmark based on a structure not necessarily centered at the pixel. The interest of these parameters will be illustrated in the Results section.

To capture the context of the landmark at different scales and distances, training images are downsized to 6 different resolutions prior to the subwindows extraction and the 6 resulting feature vectors are concatenated. For our images of size 2400×1935 pixels, these resolutions will be:

$$\frac{2400}{2^i} \times \frac{1935}{2^i} \forall i \in [0, 5].$$

Pixels of a subwindow extending beyond the image limit will be set to zero. In total, each pixel will be described by an input feature vector of size $6 \times (2W+1)^2$.

Pixel sampling scheme. Naively sampling pixels uniformly from the training images will give a very unbalanced classification problem. Indeed, each image contains 2400×1935 pixels among which only a small number belongs to the positive class. For example, for a radius $R = 2$mm, only 0.027% of the pixels (i.e., 1256) correspond to positive examples. To generate a more balanced training sample, we randomly select N pixels in each training image, where $P\%$ of these N pixels are selected among positive pixels and $100 - P\%$ are selected among negative pixels.

In addition, we constrained the image area in which the negative pixels are selected by taking into account the fact that a landmark is located in very close positions from one image to another. To confirm that, Table 1 reports the average distance of each landmark to its average position over all training images. These numbers show that each landmark is located in a very specific region of the image of radius of size between $5 - 15$mm. At the prediction stage (see below), we will use this information to constrain the search for a landmark to a given subregion of the image around the average landmark position in the training images. Therefore, it is enough to put in the training sample only pixels that belongs to this region. Negative examples in each image will be selected uniformly at random at a distance of at most 40mm around the landmark.

2.2 Classification model training

To train the pixel classifier, we will use the Extremely randomized tree algorithm [4]. This method builds an ensemble of T fully developed decision trees grown each from the original training sample (i.e., without bootstrapping). At each node, the best split is selected among k features chosen at random, where k can take its value between 1 and m, with m the total number of features. For each of the k (continuous) features, a discretization threshold is selected at random within the range of variation of that feature in the subset of observations in the

Table 1. Average distance of the landmarks to the average of their positions on the training set

Landmark	Avg. distance (mm)	STD	Landmark	Avg. distance (mm)	STD
(1)	5.47	3.16	(11)	9.73	4.69
(2)	8.00	4.46	(12)	9.50	4.61
(3)	6.79	3.58	(13)	9.29	4.69
(4)	4.91	2.33	(14)	10.29	5.32
(5)	8.66	4.15	(15)	8.89	4.48
(6)	10.37	5.18	(16)	12.10	6.17
(7)	11.78	5.84	(17)	6.06	3.17
(8)	11.86	6.01	(18)	8.55	4.39
(9)	11.90	5.97	(19)	5.32	2.43
(10)	7.73	4.12			

current tree node. The score of each pair of feature and threshold is computed and the best pair among the k is chosen to split the node. As a score measure, we use the Gini index reduction.

2.3 Landmark prediction

Let us denote by $\mu_l \in \mathbb{R}^2$ and $\Sigma_l \in \mathbb{R}^{2\times 2}$ respectively the average and the covariance matrix of the landmark positions across the training images and let us denote by σ_{x_l} and σ_{y_l} the standard deviation of its x and y positions respectively (i.e., the diagonal elements of Σ_l), also estimated from the training data. To make prediction of the landmark position with our tree-based pixel classifier, we proceed as follows:

- We randomly draw $16\sigma_{x_l}\sigma_{y_l}$ pixel positions from the following multivariate normal distribution:
$$\mathcal{N}(\mu_l, \Sigma_l)$$
- Each of the resulting pixels is classified by the tree ensemble and the final predicted landmark position is taken as the median position among the pixels that are predicted as being the landmark with the highest confidence by the tree-based model (i.e, which receives the highest number of votes for the positive class from the T trees in the ensemble).

This subsampling scheme allows to improve predictive performance by reducing the probability of generating spurious landmark predictions at irrelevant positions in the images. It also considerably speedups the algorithm with respect to a full scan of all image pixels.

3 Protocol

In this section, we describe the protocol we adopted to generate all 19 landmark detection models. First, parameters were set to some default value or optimized using ten-fold cross-validation and then a model was retrained, for each landmark and error criterion, using the optimal set of parameters.

Parameter tuning. The main method parameters are as follows:

- W, the size of the windows
- R, the distance to the interest point to decide on the training pixel output class
- t_x and t_y, the translation of the subwindows to define input features
- N, the number of pixels randomly sampled to train each landmark classification model
- The percentage P of positive examples among the N pixels
- k the number of features selected at each node in the Extremely Randomized Trees algorithm
- T, the number of trees

During parameter tuning, T was fixed to a default value of 500 and we used the suggested default value of k, which is the square root of the number of input features [4]. N was fixed to 500 and W to 8 in all our experiments. All other parameters were tuned by 10-fold cross-validation independently for each landmark and each error criterion relevant for the challenge.

The parameter tuning was done in several stages as follows:

- The optimal values of t_x and t_y were jointly tested in $\{0.8, 1.6, 3.2, 6.4, 12.8, 25.6\}$ (mm) for positive and negative translations using $R = 1$mm and $P = 33\%$.
- R was then optimized in $\{0.2, 0.5, 0.7, 1, 1.2, 1.5, 1.7, 2, 2.5, 3\}$ (mm) using $P = 33\%$ and the optimal values of t_x and t_y determined at the previous stage.
- Finally, P was optimized in $\{10, 20, 30, 33.33, 40, 50, 60, 70, 80, 90\}$ (%) with all other parameters set at their optimal values.

In total, this represents about 2000 cross-validation jobs for each criterion.

Final model training. Separate models were then retrained using all 100 training images for each landmark and error criterion using the optimal values of the parameters determined during the cross-validation. All non-optimized parameters were set similarly as during the cross-validation except the number of trees T, which was raised to 5000. Landmark predictions were then generated on the test image using the approach described in 2.3 (for each landmark and error criterion).

Software. We use the implementation of the Extremely Randomized Trees in scikit-learn [7] and our own python code for pixel and feature computation. Visual interpretation of the results was done using Cytomine [6], a generic web platform for the visualization and annotation of large-scale bioimages.

4 Results

Tables 2 and 3 report the best cross-validation performances after optimization for each criterion, respectively without and with translation. In this latter case,

Table 2. Results on all landmarks **without translation**, in terms of detection rates at various ranges of accuracy and mean euclidian distance to the landmark

Landmark	$\leq$ **2mm**	$\leq$ 2.5**mm**	$\leq$ 3**mm**	$\leq$ 4**mm**	**Eucl. Dist.**
sella (1)	87	90	93	96	1.4 ± 1.2
nasion (2)	80	86	86	91	1.8 ± 2.0
orbitale (3)	61	72	81	87	2.1 ± 1.7
porion (4)	76	86	92	96	1.6 ± 2.1
subspinale (5)	45	57	72	83	2.9 ± 2.5
supramentale (6)	68	80	86	95	1.9 ± 1.6
pogonion (7)	90	95	95	97	1.2 ± 1.4
menton (8)	95	97	98	99	0.9 ± 0.8
gnathion (9)	95	97	99	99	$1. \pm 1.2$
gonion (10)	36	46	55	69	3.8 ± 3.1
lower incisal incision (11)	83	87	93	95	1.4 ± 2.3
upper incisal incision (12)	87	89	92	94	1.6 ± 4.7
upper lip (13)	84	88	91	95	1.8 ± 2.8
lower lip (14)	84	90	94	96	2.4 ± 5.1
point pm or mn (15)	88	94	94	98	1.2 ± 1.2
soft tissue pogonion (16)	64	74	81	88	1.9 ± 1.8
posterior nasal spine (17)	84	89	94	98	$1.5 \pm 1.$
anterior nasal spine (18)	63	72	78	88	2.1 ± 1.9
articulate (19)	62	69	74	81	2.2 ± 2.3
Mean	75.37	82	86.74	91.84	1.83 ± 1.81

we also report in the table the values of t_x and t_y that give optimal performance for the 2.5mm detection rate. Note that values in these tables are optimal values over different parameter settings. They are therefore most probably optimistically biased and only provided here for information purpose. A more realistic assessment of our method performances will be done on the challenge test data.

There is a clear improvement for some landmarks by using translations. The sella point for example, is more correctly detected. We notice however that two particular points are not correctly detected, even at higher acceptance criterion: the supramentale and the gonion. Given the good results obtained on other landmarks and other inconclusive tests we have made on these two points, our conclusion is that either the dataset is not able to capture the high variability of the surrounding of these landmarks or there were some errors during the manual annotation process.

Figure 4 shows the position of the gonion on different images. It seems that the local position of the landmark does not fit the same structure on each of the images.

5 Conclusion

We showed that it was possible to accurately detect some of the landmarks using a combination of Extremely Randomized Forests and simple features. We think that given the small size of the dataset and the variance of the landmarks

Table 3. Results on all landmarks **with translation**, in terms of detection rates at various ranges of accuracy and mean euclidian distance to the landmark. Note: t_x and t_y are the best translation parameters obtained for the 2.5mm criterion

Landmark	$\leq$ **2mm**	$\leq$ **2.5mm**	$\leq$ **3mm**	$\leq$ **4mm**	**Eucl. Dist.**	t_x **(mm)**	t_y **(mm)**
sella (1)	95	96	96	97	1.21 ± 1.92	3.2	1.6
nasion (2)	78	83	86	90	1.86 ± 2.06	0	0
orbitale (3)	63	75	83	92	2.06 ± 1.50	0.8	-6.4
porion (4)	77	86	92	97	1.53 ± 1.22	0	0
subspinale (5)	54	63	71	83	2.78 ± 2.20	0	1.6
supramentale (6)	71	78	86	95	1.84 ± 1.56	-1.6	-0.8
pogonion (7)	89	94	97	99	1.21 ± 1.30	0	0
menton (8)	94	97	98	100	0.94 ± 0.80	0.8	-0.8
gnathion (9)	97	99	99	100	0.91 ± 0.69	-3.2	-0.8
gonion (10)	38	48	56	66	3.76 ± 2.85	1.6	1.6
lower incisal incision (11)	89	92	95	97	1.44 ± 2.35	-1.6	1.6
upper incisal incision (12)	88	92	95	97	1.29 ± 3.27	3.2	-6.4
upper lip (13)	84	89	93	95	1.56 ± 2.08	0	-0.8
lower lip (14)	87	93	96	99	1.45 ± 2.36	-0.8	-0.8
point pm or mn (15)	88	92	95	98	1.19 ± 1.07	1.6	1.6
soft tissue pogonion (16)	67	75	83	91	1.94 ± 1.80	-0.8	-1.6
posterior nasal spine (17)	83	90	95	98	1.38 ± 1.06	0.8	0.8
anterior nasal spine (18)	67	78	84	91	2.01 ± 1.56	-3.2	0
articulate (19)	65	74	79	86	2.28 ± 2.06	1.6	3.2
Mean	77.58	83.89	88.37	93.21	1.72 ± 1.77		

between the images, these results are promising in comparison to existing algorithms. The main advantage of our approach with respect to existing works is its simplicity and efficiency. High level features such as Zernike moments can accurately describe an image or a window, but they are slow to compute, which could be detrimental in some applications.

We think that the main problem for our machine learning based approach is the lack of data: for some landmarks, 100 images does not seem to be enough to grasp the variability of the possible landmark structures. Moreover, it seems that some landmarks do not especially correspond to specific shapes or structures, but more to positions or intersections.

We believe that further improvement could probably be obtained by taking into account the relative positions of the 19 landmarks either directly during the training or during the prediction stage. We have made some experiments in this direction but we were not able to improve with respect to the results reported here. Further improvement could also be brought by considering different values of W at each of the different resolutions or different feature extraction methods.

Acknowledgment

The authors thank the GIGA Bioinformatics platform and the SEGI for providing computing resources. Rémy Vandaele is funded by a Televie grant from the Belgian National Science Fundation (F.N.R.S.).

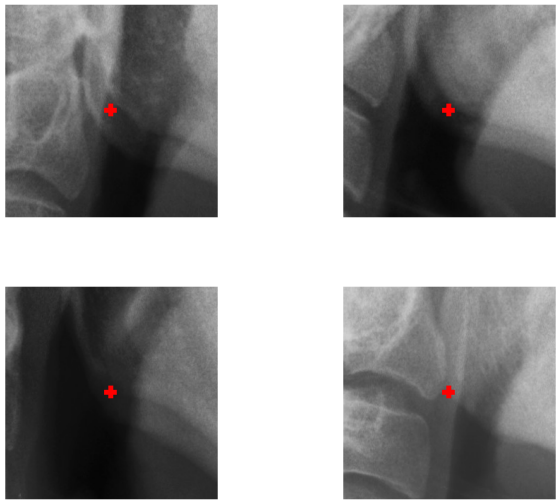

Fig. 1. Gonion surroundings for training set images 10,20,30 and 40. In red, the position of the gonion landmark

References

1. L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
2. A. Criminisi and J. Shotton. *Decision Forests for Computer Vision and Medical Image Analysis*. Springer, 2013.
3. A. Criminisi, J. Shotton, D. Robertson, and E. Konukoglu. Regression forests for efficient anatomy detection and localization in ct studies. In *Medical Computer Vision. Recognition Techniques and Applications in Medical Imaging*, pages 106–117. Springer, 2011.
4. P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine learning*, 63(1):3–42, 2006.
5. A. Kaur and C. Singh. Automatic cephalometric landmark detection using zernike moments and template matching. *Signal, Image and Video Processing*, pages 1–16, 2013.
6. R. Marée, L. Rollus, G. Louppe, O. Caubo, N. Rocks, S. Bekaert, D. Cataldo, and L. Wehenkel. A hybrid human-computer approach for large-scale image-based measurements using web services and machine learning. In *Proceedings IEEE International Symposium on Biomedical Imaging*. IEEE, 2014.
7. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
8. O. Stern, R. Marée, J. Aceto, N. Jeanray, M. Muller, L. Wehenkel, and P. Geurts. Automatic localization of interest points in zebrafish images with tree-based methods. In *Pattern Recognition in Bioinformatics*, pages 179–190. Springer, 2011.