Fully-Automatic Landmark detection in Skull X-Ray images

Cheng Chen[1], Ching-Wei Wang[2,3], Cheng-Ta Huang[2], Chung-Hsing Li[3,4], and Guoyan Zheng[1]

[1] Institute for Surgical Technology and Biomechanics, University of Bern, Switzerland cheng.chen@istb.unibe.ch, guoyan.zheng@ieee.org
[2] Graduate Institute of Biomedical Engineering, National Taiwan University of Science and Technology, Taiwan
[3] Dental Department, National Defense Medical Center, Taiwan
[4] Orthodontics and Pediatric Dentistry Division, Dental Department, Tri-Service General Hospital, Taiwan

Abstract. In this paper, we present our new method for fully-automatic landmark detection on X-ray images. To detect landmarks, we estimate the displacements from some randomly sampled image patches to the (unknown) landmark positions, and then we integrate these predictions via a voting scheme. Different from other methods, we jointly estimate the displacements from all patches together in a data driven way, by considering not only the training data but also geometric constraints on the test image. The displacements estimation is formulated as a convex optimization problem that can be solved efficiently. We validate our method on a dataset of skull X-ray image containing 100 training images and 100 test images. We get an average detection error of 2.8 mm.

1 Introduction

In clinical practice, X-ray radiography is widely used for various purposes due to its convenience and low cost. Detecting key points (landmarks) on X-ray images benefits many applications. Traditionally, landmark detection is seldom done in clinical practice due to its difficulty. In cases where it is ever done, it is carried out manually by doctors, which is both time-consuming and error-prone. Therefore, fully-automatic techniques will immediately make this traditionally useful but difficult task widely applicable.

In [1], we proposed a new method for this task. We estimate the displacements from a set of randomly sampled local image patches to the landmark based on patch appearance, and the individual predictions are then combined in a voting scheme to produce the predicted landmark position. In previous methods, the displacement from each patch to the landmark is estimated independently using a pre-trained model. Our method is fundamentally different, as we jointly estimate the displacements from all patches to landmarks together in a data-driven way. This joint estimation scheme allows us to exploit the mutual interactions among the displacements that are being estimated by considering the geometric relations

between the patches in the test image. Combining the information from training data and the geometry constraints, our displacement estimation method achieves better accuracy.

We tested our method on a dataset of skull X-ray images involving 19 landmarks. The dataset is devided into 100 training images and 100 test images (with no overlapping). We get an average detection error of 29 pixels, with 40%, 52% 62% and 78% of landmarks detected within an error less than 2.0mm, 2.5mm, 3.0mm and 4.0mm, respectively.

2 Related Work

In recent literature, there has been a considerable amount of work in landmark detection. Some methods utilize low-level image features such as gradients and edges [2]. This type of methods often suffers from the large appearance variation and image noise. To alleviate this problem, some similar methods such as [3, 4] incorporate the topological constraints in a model-based way.

To overcome the challenge of appearance variation, some machine learning based methods have been proposed. For example, in [5], a particle filter-based approach is first used to determine the morphological parameters, and then a belief propagation based approach is used to extract contours from multiple calibrated X-ray images. [6] introduce the so-called shape regression machine to segment in real time the left ventricle endocardium from an echocardiogram of an apical four chamber view. [7] use marginal space learning for localizing the heart chambers, and then estimate the 3D shape through learning-based boundary delineation.

In recent years, random forest (RF) based methods are becoming more and more popular [8–10]. The basic idea is to estimate the displacement displacements from a set of randomly sampled image patches to the (unknown) landmark position by RF regression. The landmark position is estimated by a voting scheme considering the individual estimations from all the patches. In this paper, we follow the framework of predicting relational displacements from image patches. However, instead of using RF, our method improves the displacement prediction by a data-driven approach.

3 Landmark Detection Algorithm

In this section we present our landmark detection algorithm. For more details, please refer to [1].

3.1 Basic Idea

The overview of our landmark detection framework is given in Fig. 1. Please note both this paper and [11] use the same framework but different detection algorithms. In the training step, as shown in (a), a rectangular image patch is

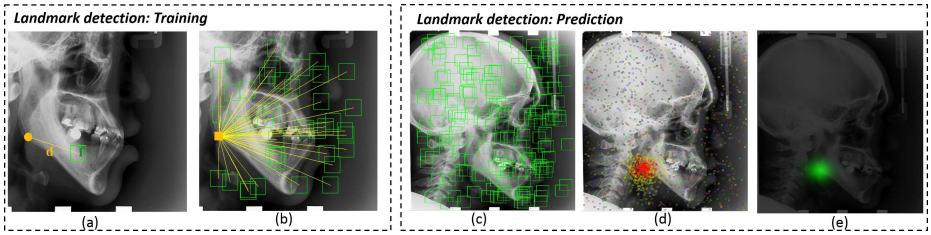

Fig. 1. Overview of our landmark detection algorithm. During training stage (a)(b), a set of patches are sampled around the ground-truth landmark position. During testing stage, given an image, a set of patches are randomly sampled (c), each patch makes a prediction of the landmark position (d), and then the predictions are aggregated to produce a response image (e).

randomly sampled around the ground-truth landmark position, with f denoting the visual feature of the patch, and d denoting the displacement from this patch to the landmark position. In the same way, we randomly sample a number of patches around the ground-truth landmark position, as shown in (b). The features and corresponding displacements of all these training patches constitute the training data. Then, in the prediction step, given a new image, we also randomly sample a number of image patches, as in (c). Since now we do not know the landmark position, these test patches are sampled everywhere in the image[5]. The visual features of these patches will be calculated, and based on their features, the corresponding displacements with regard to the (unknown) landmark position can be estimated. In this way, each patch makes a vote on the landmark prediction as in (d), where each vote contains a position (depicted by dots) and uncertainty (color-coded). Then, from these votes, we construct the *response image* as in (e), which can be viewed as the probability of the landmark position on every image location.

The crucial part of the procedure presented above is the estimation of displacements for the test patches, which is formally described in the following.

3.2 Notations

Training step. Assume that $\tilde{\mathbf{x}} \in \mathbb{R}^2$ is the ground-truth landmark position. We randomly sample a number of patches around $\tilde{\mathbf{x}}$. For the kth patch, we denote $\tilde{\mathbf{c}}_k \in \mathbb{R}^2$ as its center position, $\tilde{\mathbf{f}}_k \in \mathbb{R}^{d_f}$ as its visual feature, and $\left(\tilde{\mathbf{d}}_k\right)_{GT} = \tilde{\mathbf{x}} - \tilde{\mathbf{c}}_k \in \mathbb{R}^2$ as its ground-truth (GT) displacement to the landmark. In total, we sample $\tilde{K}$ patches over all the training images, and we denote $\tilde{\mathbf{F}} = \left[\tilde{\mathbf{f}}_1, ..., \tilde{\mathbf{f}}_{\tilde{K}}\right] \in \mathbb{R}^{d_f \times \tilde{K}}$ as the training feature matrix, and $\left(\tilde{\mathbf{D}}\right)_{GT} = \left[\left(\tilde{\mathbf{d}}_1\right)_{GT}, ..., \left(\tilde{\mathbf{d}}_{\tilde{K}}\right)\right] \in \mathbb{R}^{2 \times \tilde{K}}$, as the ground-truth displacement matrix.

[5] Or the relevant ROI (region of interest) of the image. See the experiment section for details of our multi-resolution implementation.

Prediction (test) step. Given a new image, we randomly sample K patches, where $\mathbf{c}_k \in \mathbb{R}^2$ and $\mathbf{f}_k \in \mathbb{R}^{d_f}$ are the center position and the visual feature of the kth patch. We denote $\mathbf{F} = [\mathbf{f}_1, ..., \mathbf{f}_k] \in \mathbb{R}^{d_f \times K}$ as the test feature matrix.

We want to estimate $\{\mathbf{d}_k\}_{k=1...K}$, the displacement from each patch to the landmark. If we denote $\mathbf{D} = [\mathbf{d}, ..., \mathbf{d}_K] \in \mathbb{R}^{2\times K}$, our goal is to estimate $\mathbf{D}$.

3.3 Objective Function

First, we construct a compound displacement matrix which contains jointly the training displacements and the test displacements to be estimated:

$$\hat{\mathbf{D}} = \begin{bmatrix} \tilde{\mathbf{D}} & \mathbf{D} \end{bmatrix} = \left[\tilde{\mathbf{d}}_1, ..., \tilde{\mathbf{d}}_{\tilde{K}}, \mathbf{d}_1, ..., \mathbf{d}_K\right] \in \mathbb{R}^{2\times(\tilde{K}+K)} \tag{1}$$

The left part (the first $\tilde{K}$ columns) of $\hat{\mathbf{D}}$ contains the displacements in the training images, and the right part (the last K columns) is the displacements in the test image. Note that we can write $\tilde{\mathbf{D}} = \hat{\mathbf{D}}\mathbf{P}$ and $\mathbf{D} = \hat{\mathbf{D}}\mathbf{Q}$ by defining appropriate (0,1) matrices $\mathbf{P}$ and $\mathbf{Q}$ which select corresponding columns.

Treating $\hat{\mathbf{D}}$ as a variable, we design an objective with regard to $\hat{\mathbf{D}}$:

$$E(\hat{\mathbf{D}}) = E_g(\hat{\mathbf{D}}) + \alpha E_f(\hat{\mathbf{D}}) + \beta E_p(\hat{\mathbf{D}}) \tag{2}$$

Please note that although we are ultimately interested in estimating the displacements of test patches $\mathbf{D}$, our objective function is defined on $\hat{\mathbf{D}}$, which is the combination of training and test displacements. In this way we can embed the relations between training and test data into our objective function. After we get the optimal $\hat{\mathbf{D}}$, the optimal $\mathbf{D}$ is simply given by $\mathbf{D} = \hat{\mathbf{D}}\mathbf{Q}$. Below we define each term in the objective function.

Ground-truth Discrepancy $E_g(\hat{\mathbf{D}})$. The left part of $\hat{\mathbf{D}}$ should be close to the ground-truth displacements in the training data, which is encoded in $\left(\tilde{\mathbf{D}}\right)_{GT}$. Therefore, we want to minimize the Ground-truth Discrepancy:

$$E_g(\hat{\mathbf{D}}) = \frac{1}{2\tilde{K}} \left\| \hat{\mathbf{D}}\mathbf{P} - \left(\tilde{\mathbf{D}}\right)_{GT} \right\|_F^2 \tag{3}$$

where $\|.\|_F$ is the Frobenius norm.

Feature Propagation Discrepancy $E_f(\hat{\mathbf{D}})$. First, we construct a compound feature matrix $\hat{\mathbf{F}} = \begin{bmatrix} \tilde{\mathbf{f}} & \mathbf{f} \end{bmatrix} \in \mathbb{R}^{d_f \times (\tilde{K}+K)}$. Now, each column of $\hat{\mathbf{F}}$ is the feature of a (training or test) patch, and the corresponding column of $\hat{\mathbf{D}}$ is the displacement vector (to all landmarks) of that patch. We denote $\|\text{col}_i(\hat{\mathbf{F}}) - \text{col}_j(\hat{\mathbf{F}})\|_{L2}$ as the $L2$ feature distance of a pair of patches (i, j), where $\text{col}_i()$ denotes the ith column. From all pairwise distances, we construct a binary affinity matrix $\mathbf{S} \in \{0, 1\}^{(\tilde{K}+K)(\tilde{K}+K)}$, where $s_{ij} = 1$ if and only if the ith and the jth patches are mutually ρ nearest neighbors ($\rho = 10$ in this paper) in the feature space. Note that the edges in the affinity matrix might link two training patches, two test patches, or a training patch and a test patch.

For every pair of patches (i, j), if they are similar in the feature space, their displacements to landmarks should also be similar. We define the Feature Propagation Discrepancy $E_f(\hat{\mathbf{D}})$ as the violation from this assumption:

$$E_f(\hat{\mathbf{D}}) = \frac{1}{2\sum_{i\neq j}} \sum_{i\neq j} s_{ij} \left\| \mathrm{col}_i(\hat{\mathbf{D}}) - \mathrm{col}_j(\hat{\mathbf{D}}) \right\|_{L2}^2 \tag{4}$$

For each pair of patches, E_f introduces a high penalty if the two patches are similar in the feature space (i.e. $s_{ij} = 1$) but their displacements are very different (i.e. $\left\| \mathrm{col}_i(\hat{\mathbf{D}}) - \mathrm{col}_j(\hat{\mathbf{D}}) \right\|_{L2}$ is large). If we construct $\mathbf{M}$ as the (trace normalized) laplacian matrix [12] of $\mathbf{S}$, E_f can be compactly written as:

$$E_f(\hat{\mathbf{D}}) = \mathrm{Tr}\left(\hat{\mathbf{D}}\mathbf{M}\hat{\mathbf{D}}^\top\right) \tag{5}$$

In short, this term favors the consistency between feature proximity and displacement proximity. In this way, the ground-truth displacements are propagated to the test data via the links between training and test patches.

Patch Offset Penalty $E_p(\hat{\mathbf{D}})$.
Each column of $\mathbf{D}$ is the displacements from a single patch in the test image to the landmark position. The subtraction of two columns can be written as $\mathrm{col}_i(\mathbf{D}) - \mathrm{col}_j(\mathbf{D}) = \mathbf{D}(\mathbf{e}_i^K - \mathbf{e}_j^K) = [\mathbf{d}_i - \mathbf{d}_j]$, where $\mathbf{e}_i^K$ is a K dimensional column vector whose ith element is 1 and all other elements are 0s. it is not difficult to see that $\mathbf{d}_i - \mathbf{d}_j = \mathbf{c}_j - \mathbf{c}_i$, because $(\mathbf{d}_i, \mathbf{d}_j)$ form a triangle with the same edge $\mathbf{c}_j - \mathbf{c}_i$. Therefore, we impose a penalty $E_p^{i-j}(\mathbf{D}) = \left\|\mathbf{D}(\mathbf{e}_i^K - \mathbf{e}_j^K) - \bar{\mathbf{c}}_{j-i}\right\|_F^2$, where $\bar{\mathbf{c}}_{j-i} = \mathbf{c}_j - \mathbf{c}_i$. We can include a penalty for each pair (i, j) of columns. For efficiency reasons, we eliminate redundancies and use $K - 1$ pairs:

$$E_p(\hat{\mathbf{D}}) = \frac{1}{2K} \sum_{i=1}^{K-1} E_p^{i-(i+1)}(\mathbf{D}) = \frac{1}{2K} \left\| \hat{\mathbf{D}}\mathbf{Q}\mathbf{U} - \bar{\mathbf{C}} \right\|_F^2 \tag{6}$$

where $\mathbf{U} = \left[\mathbf{e}_1^K - \mathbf{e}_2^K, ..., \mathbf{e}_{K-1}^K - \mathbf{e}_K^K\right]$ and $\bar{\mathbf{C}} = \left[\, \bar{\mathbf{c}}_{2-1} \; ... \; \bar{\mathbf{c}}_{K-(K-1)} \,\right]$.

3.4 Optimization

Substituting Eqs. (3),(5) and (6) into Eq. (2), we get the final objective function. We can prove that Eq. (2) is convex, and therefore to find the global optimum, we need to solve the equation:

$$\partial E(\hat{\mathbf{D}}) \Big/ \partial \hat{\mathbf{D}} = \hat{\mathbf{D}}\mathcal{A} + \mathcal{G} = \mathbf{0} \tag{7}$$

where $\mathcal{A} = \frac{1}{LK}\mathbf{P}\mathbf{P}^\top + \frac{2\alpha}{L}\mathbf{M} + \frac{\beta}{LK}\mathbf{Q}\mathbf{U}\mathbf{U}^\top\mathbf{Q}^\top$, and $\mathcal{G} = -\frac{(\tilde{\mathbf{D}})_{GT}\mathbf{P}^\top}{LK} - \frac{\beta\bar{\mathbf{C}}\mathbf{U}^\top\mathbf{Q}^\top}{LK}$. The optimal solution is given by $\hat{\mathbf{D}} = -\mathcal{G}\mathcal{A}^{-1}$.

Landmark	mean (mm)	std. (mm)	<2.0mm(%)	<2.5mm(%)	<3.0mm(%)	<4.0mm(%)
AVER.	**28.1**	**18.7**	**43.9**	**54.6**	**64.2**	**78.4**

Table 1. The statistical result of our method on the 100 test images.

3.5 Constructing Response Image

After we find the optimum $\hat{\mathbf{D}}$, we have $\mathbf{D} = \hat{\mathbf{D}}\mathbf{Q}$, and $\{\mathbf{c}_k + \mathbf{d}_k\}_{k=1...K}$ will be the set of votes for the landmark position. We write $\mathbf{v}_k = \mathbf{c}_k + \mathbf{d}_k$ as the position vote made by the kth patch. For each vote, there is also an uncertainty $\mathbf{\Sigma}_k$, which is calculated as the (diagonal) variance of the training displacements that are linked to the kth test patch when we calculated the feature propagation discrepancy $E_f(\hat{\mathbf{D}})$ in Section 3.3. The next step is to calculate the probability of landmark on different image locations, from the votes $\{(\mathbf{v}_k, \mathbf{\Sigma}_k)\}_{k=1...K}$. We view each vote $(\mathbf{v}_k, \mathbf{\Sigma}_k)$ as a Gaussian distribution $G(.|\mu, \mathbf{\Sigma})$ with mean $\mathbf{v}_k$ and variance $\mathbf{\Sigma}_k$. Then, the probability of landmark at an image coordinate (x, y) is given by accumulating the contribution of all votes on this image location:

$$I(x, y) = \sum_{k=1}^{K} G\left((x, y)|\mathbf{v}_k, \mathbf{\Sigma}_k\right) \tag{8}$$

$I(x, y)$ is viewed as an image function which is called *response image* of the landmark, as in Fig. 1 (e). And the final point estimate of the landmark position is the image position where the response image gets its maximum value (mode).

4 Experiments

4.1 Data and Implementation details

We validated our method on a dataset of 200 skull X-ray images involving 19 landmarks. The dataset is divided into 100 training images and 100 test images. The pixel spacing of the images is 0.1 mm/pixel in each dimension.

To improve the efficiency, we adopt a multi-scale strategy with two scale levels. In the first level the image is subsampled to its 1/3 size in each dimension, and the second level operates on the original image. The result of the first level is propagated to the second level as initialization, where the landmarks are detected by sampling patches only in a limited region around the initial position. At the first level where no initialization is available, the landmarks are detected by sampling patches all through the image. In this way, we combine the global detection at the first level and local detection at the higher levels, which achieves high accuracy without exploding the computation time.

In each scale level, we use the same parameters as follows: For each landmark, we sample $\tilde{K} = 2000$ training patches and $K = 1000$ test patches. For the visual feature of the patches, we use multi-level HoG (Histogram of Oriented Gradient) feature [13] with block sizes 1×1 and 2×2. Each block is divided into 2×2 cells

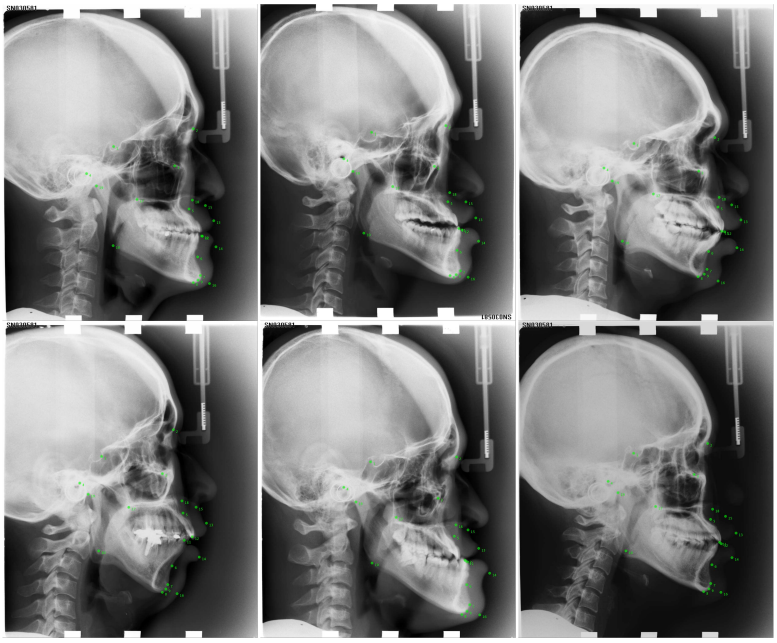

Fig. 2. Some qualitative result of our method.

and for each cell an 18 dimensional HoG feature is extracted by histogramming the gradient direction of each pixel. Therefore, our original feature dimension is $d_f = 360$.

4.2 Results

Fig. 2 shows qualitative results on several randomly chosen test images, where we plot the detected landmark on the image as green dots.

Table 1 shows the statistical result of our method. We can see that we ahieve an average detecting error as 28.1mm with standard deviation of 18.7mm. The percentage of detections whose errors are below 2.0mm, 2.5mm, 3.0mm and 4.0mm are 43.9%, 54.6%, 64.2% and 78.4%, respectively.

Note that we detect the 19 landmarks independently of each other. As a further improvement, we could use some high-level model on top of the output from our method to regularize the inter-relations between the landmarks, such as the sparse shape model as in [1], or the MRF based model as in [10].

5 Conclusions

We applied our new method for landmark detection on the dataset of skull X-ray images. Our method works by jointly estimating the image displacements of test

patches using the training data and also the geometric information on the test image itself. The key contribution is the exploitation of the inter-patch relations to impose the geometric regularizations on the image displacements that are being estimated. We formulate our problem as a convex objective function which can be solved efficiently. The validation on the skull X-ray image dataset involving 19 landmarks shows that we achieve an average detection error of 29.2mm.

References

1. C. Chen, W. Xie, J. Franke, P.A. Grutzner, L.-P. Nolte, G. Zheng, Automatic X-ray landmark detection and shape segmentation via data-driven joint estimation of image displacements. Medical Image Analysis, 18(3): 487-499 (2014)
2. Cristinacce, D., Cootes, T.: Automatic feature localization with constrained local models. Pattern Recognition, 41(19), 3054-3067 (2008)
3. Bergtholdt, M., Kappes, J., Schmidt, S., Schnrr., C.: A study of parts-based object class detection using complete graphs. International Journal of Computer Vision 87, 93-117 (2010)
4. Schmidt, S., Kappes, J., Bergtholdt, M., Pekar, V., Dries, S., Bystrov, D., Schnrr, C.: Spine detection and labeling using a parts-based graphical model, in: IPMI, pp. 122-133 (2007)
5. Dong, X., Zheng, G.:. Automatic extraction of proximal femur contours from calibrated x-ray images using 3d statistical models: an in vitro study. The International Journal of Medical Robotics and Computer Assisted Surgery 5, 213-222 (2009).
6. Zhou, S.K., Comaniciu, D.: Shape regression machine, in: IPMI, pp. 13-25 (2007).
7. Zheng, Y., Barbu, A., Georgescu, B., Scheuering, M., Comaniciu, D.: Four-chamber heart modeling and automatic segmentation of 3-D cardiac CT volumes using marginal space learning and steerable features. IEEE T. Med. Imaging, 27(11), 1668-1681 (2008)
8. Criminisi, A., Shotton, J., Robertson, D., Konukoglu, E.: Regression forests for efficient anatomy detection and localization in CT studies. In: proceedings of MICCAI 2010 workshop in Medical Computer Vision: recognition techniques and applications in medical imaging, pp. 106-117 (2010)
9. Lindner, C., Thiagarajah, S., Wilkinson, J.M., Wallis, G.A., Cootes, T.F.: Fully automatic segmentation of the proximal femur using random forest regression voting. IEEE Trans. Med. Imaging 32, 1462-1472 (2013).
10. Donner, R., Menze, B.H., Bischof, H., Langs, G.: Global localization of 3d anatomical structures by pre-filtered hough forests and discrete optimization. Medical Image Analysis, 17(8): 1304-1314 (2013).
11. Chu, C., Chen, C., Wang, C-W., Huang, C-T., Li, C-H, Nolte, LP, Zheng, G.: Fully Automatic Cephalometric X-Ray Landmark Detection Using Random Forest Regression and Sparse shape composition. submitted to Automatic Cephalometric X-ray Landmark Detection Challenge 2014.
12. Kokiopoulou, E., Chen, J., Saad, Y.: Trace optimization and eigenproblems in dimension reduction methods. In: Numerical Linear Algebra with Applications, 18(3), 565-602 (2011)
13. Dalal, N., Triggs, B., Histograms of oriented gradients for human detection. In: CVPR (2005)