Automatic Collimation Detection in Digital Radiographs with the Directed Hough Transform and Learning-Based Edge Detection

Liang Zhao[2], Zhigang Peng[1]([✉]), Klaus Finkler[4], Anna Jerebko[4], Jason J. Corso[3], and Xiang (Sean) Zhou[1]

[1] Siemens Medical Solutions USA, Inc., Malvern, USA
zhigang.peng@siemens.com
[2] Computer Science and Engineering, University at Buffalo SUNY, Buffalo, USA
[3] Electrical Engineering and Computer Science, University of Michigan,
Ann Arbor, USA
[4] Siemens Healthcare, Erlangen, Germany

Abstract. Collimation is widely used for X-ray examinations to reduce the overall radiation exposure to the patient and improve the contrast resolution in the region of interest (ROI), that has been exposed directly to X-rays. It is desirable to detect the region of interest and exclude the unexposed area to optimize the image display. Although we only focus on the X-ray images generated with a rectangular collimator, it remains a challenging task because of the large variability of collimated images. In this study, we detect the region of interest as an optimal quadrilateral, which is the intersection of the optimal group of four half-planes. Each half-plane is defined as the positive side of a directed straight line. We develop an extended Hough transform for directed straight lines on a model-aware gray level edge-map, which is estimated with random forests [1] on features of pairs of superpixels. Experiments show that our algorithm can extract the region of interest quickly and accurately, despite variations in size, shape and orientation, and incompleteness of boundaries.

1 Introduction

In medical X-ray examinations, radiographers often adopt collimation to avoid unnecessary exposure to patients and improve the image quality. It is important to detect the non-collimated region for improving the visual quality in the diagnosis environment and for the processing of the image, e.g. image compression. Nevertheless, collimation detection remains a challenging problem due to the large variability in shape and appearance across collimated images. In this study, the region of interest (ROI) is the non-collimated region in the X-ray image, which is a quadrilateral caused by rectangle collimators.

There have been many works using 2D boundary detection technique for collimation detection [2,3]. However, all these methods used an unsupervised model for edge detection. Because of the large variability of the collimated images
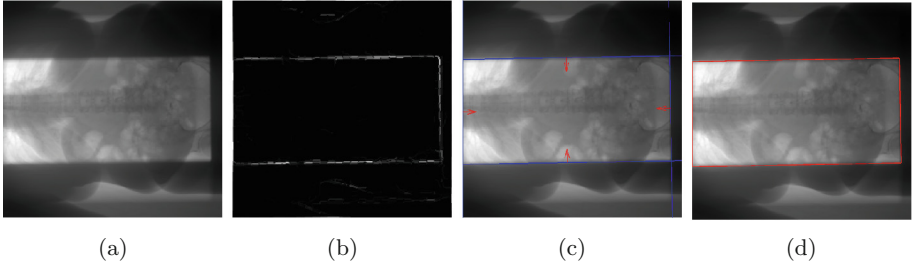
| (a) | (b) | (c) | (d) |

Fig. 1. An example of our algorithm: (a) Input image; (b) Edge map; (c) The optimal group of four directed straight lines; (d) Detected ROI.

and the overlap of the pixel-level feature distribution between the ROI and the collimation field, these method are not reliable. In Kawashita et al. [4], the authors detect the ROI with a plane detection Hough transform. But this method works only when the ROI is a rectangle and is not as large as the whole image. In Mao et al. [5], the authors proposed a multi-view learning-based method combining region and corner detection. The accuracy of this method highly depends on the region detection with a pixel-level two-class classification. As this method makes a classification on each pixel, the accuracy is also limited and compromises efficiency. In this paper, we develop our method based on two simple observations. First, if two pixels are near to each other, one is in the ROI and one is not, the difference of their intensity is often relatively large. Second, along a piece of boundary between the ROI and the collimation field, the directions of gradient vectors are often from the collimation field to the ROI.

The first observation indicates that, if we over-segment the image with both location and appearance information, e.g. SLIC [6], most pixels of the boundary of the ROI would be on the boundaries of superpixels. The second observation indicates that, compared with undirected straight lines, straight lines with positive normal directions specified can make stricter shape constraints for the ROI detection in X-ray images.

Hence, we proposed a novel algorithm based on superpixel-level learning-based edge estimation and a directed Hough transform. We estimate the edge strength for pixels with a random forests approach on pairs of neighbouring superpixels (Fig. 1b). As the edge map is estimated with the training data, it is robust and accurate. By computing the edge map on the superpixel level, we decrease the number of samples for learning-based classification, take advantage of superpixel-level features and decrease the number of pixels visited in the Hough transform. In Andrews and Hamarneh's work [7], the authors also used learning-based method for boundary detection. However, their method makes classifications for each boundary candidates on the pixel-level and our method only makes classifications on pairs of neighboring superpixels, which should be more efficient.

We define directed straight lines by specifying their normal vector, and extend the classical Hough transform to detect directed straight lines with the gradient

vector field and the learning-based edge map. We define positive and negative half-planes for directed straight lines. And we regard the quadrilateral ROI as the intersection of four positive half-planes of the optimal group of four directed lines, as shown in Fig. 1c and d. By setting special values in the edge map and the vector field for the pixels on the four sides of the image, our method works even if the ROI is the whole image.

2   Learning-Based Edge Estimation

Given an X-ray image I, we compute its edge map I_e with the following four steps (Fig. 2).

1. We make an over-segmentation on the image with SLIC [6], resulting in a set of superpixels, S.
2. For each pair of neighbouring superpixels, we compute its probability of belonging to each of the three given classes with random forests (See Sect. 2.1 for details).
3. For each superpixel, we compute its probability of being in the ROI.
4. For each pixel $p \in I$, we compute its edge strength $I_e(p)$ based the computed superpixel-level probabilities.

As our algorithm is based on the assumption that most pixels on the boundary of the ROI are on the boundaries of some superpixels in S, we use the algorithm SLIC (Simple Linear Iterative Clustering) [6] to make the over-segmentation with pixel-level features, including intensity and texture information, which efficiently produces smooth regular-sized superpixels. We select this algorithm to guarantee our assumption above and make the features of different superpixels comparable.
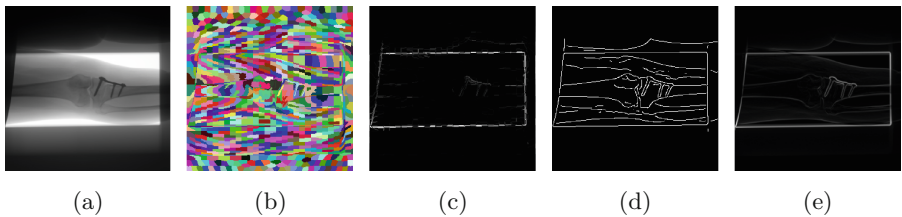

Fig. 2. An example of our edge estimation algorithm: (a) Input image; (b) superpixels; (c) Edge map with our method; (d) Canny edge map [8]; (e) gradient magnitude

2.1   Superpixel-Level Probability

We use a three-class random forests classifier to estimate the probability of each pair of neighbouring superpixels. The set of classes is $\{L_0, L_1, L_2\}$, which represents that 0, 1 or 2 superpixels of this pair are in the ROI, respectively. Let $Pr_{(P,Q)}(L)$ be the probability of the pair of superpixels, (P,Q), belonging

to the class Ł. The features used in the classifier include the averages and the standard deviation of pixel-level features, such as intensity and gradients, of the pixels contained by either of the two superpixel, and properties of shape. By comparing the average of the intensity of P and Q, and rearranging their features, we guarantee that (P,Q) and (Q,P) have the same feature value for the random forests classifier and $Pr_{(P,Q)}(\text{Ł}) = Pr_{(Q,P)}(\text{Ł})$, $\text{Ł} = \text{Ł}_0$, Ł_1, Ł_2.

Let $Pr(P)$ be the probability for the superpixel P of being in the ROI, which is defined as: $Pr(P) = \frac{\sum_{Q \in \mathbf{N}(P)} Pr_{(P,Q)}(\text{Ł}_2)}{|\mathbf{N}(P)|}$, where $\mathbf{N}(\cdot)$ is the neighbourhood of a superpixel, $|\cdot|$ is the size of a set.

2.2   Pixel-Level Probability

Let $B(P,Q)$ be the common boundary of a pair to superpixels (P,Q), l_{four} be the set of pixels on the four sides of the image, such that $l_{four} = \{p\|p_x = 1|p_x = M|p_y = 1|p_y = N\}$, where $M \times N$ is the size of I.

Note that, given a perfect over-segmentation S of I, such that each superpixel P is either exactly in the ROI or exactly not, a pixel p is on the boundary of the ROI if and only if it is in one of the following two situations:

– $p \in B(P,Q)$, one of P and Q is in the ROI, the other is not.
– $p \in l_{four}$, $p \in P$, P is in the ROI

Hence we estimate the strength of edge $I_e(p)$ with the probability of pixel p being on the boundary:

$$I_e(p) = \begin{cases} \alpha Pr(P) & \text{if } p \in P \text{ and } p \in l_{four} \\ Pr_{(P,Q)}(\text{Ł}_1) & \text{if } p \in B(P,Q) \text{ and } p \notin l_{four} \\ 0 & \text{otherwise} \end{cases} \qquad (1)$$

where, α is positive constant parameter.

3   Directed Hough Transform

3.1   Classical Hough Transform

Applying the classical Hough transform (HT) to the non-negative edge map I'_e of an image I results in an 2D accumulator array $C(\rho_k, \theta_l)$, that represents the sum of edge strength (I'_e) of points satisfying the linear equation $\rho_k = p_x \cos \theta_l + p_y \sin \theta_l$, where p is a pixel in the image and (p_x, p_y) is its coordinate. Local maxima of C can be used to detect straight lines in the image. $C(\cdot)$ is defined as: $C(\rho_k, \theta_l) = \sum_{p:\ p_x \cos \theta_l + p_y \sin \theta_l - \rho_k = 0} I'_e(p)$. Where, I'_e can be binary, e.g. Canny edges, or grey-level, e.g. gradient magnitude. To compute $C(\cdot)$, for each pixel p, such that $I'_e(p) > 0$, the Hough transform algorithm calculates the parameters (ρ, θ) of lines at p, and increments the value of the corresponding bins in $C(\cdot)$ by $I'_e(p)$ [9].

3.2 Directed Straight Line

Given the coordinate (p_{0x}, p_{0y}) of a pixel p_0 and a normal vector $\boldsymbol{n_0}$, a straight line passing through p_0 is defined with a linear equation:

$$p_x \cos \theta_0 + p_y \sin \theta_0 - \rho_0 = 0 \tag{2}$$

where θ_0 is the angle between $\boldsymbol{n_0}$ and the positive X-axis, ρ_0 is defined as $p_{0x} \cos \theta_0 + p_{0y} \sin \theta_0$. In this definition, $\theta_0 \in [0, \pi)$.

In this paper, we use the same equation (Eq. 2) to define a directed straight line $l_{p_0, \boldsymbol{n_0}}$ by measuring the angle θ_0 counter-clockwise from the positive X-axis to $\boldsymbol{n_0}$. In this way, a point with two opposite normal vectors define two different directed lines passing through the same points. $\theta_0 \in [0, 2\pi)$, and θ_0 is in one-to-one correspondence to normal vectors $\boldsymbol{n_0}$, $\boldsymbol{n_0} = (\cos \theta_0, \sin \theta_0)$.

We define the positive and negative half-plane of $l_{p_0, \boldsymbol{n_0}}$ with two linear inequalities, respectively:

$$H^+(l_{p_0, \boldsymbol{n_0}}): \ p_x \cos \theta_0 + p_y \sin \theta_0 - \rho_0 > 0 \tag{3}$$

$$H^-(l_{p_0, \boldsymbol{n_0}}): \ p_x \cos \theta_0 + p_y \sin \theta_0 - \rho_0 < 0 \tag{4}$$

Given a vector field $\boldsymbol{W} = (U, V)$ for the image I, a pixel q is a supporting pixel of the directed line $l_{p_0, \boldsymbol{n_0}}$, if and only if $(q_x, \ q_y)$ satisfies Eq. 2 and $|\angle(\boldsymbol{W}(q), \boldsymbol{n_0})| < \pi/2$. It is easy to see, the directions of the vectors of supporting pixels are from the negative half-plane to the positive half-plane. We define $\boldsymbol{W}$ as the supporting vector field of the image I and estimate the strength of directed straight lines with its supporting pixels in Sect. 3.3.

3.3 Directed Hough Transform with Gradient Vectors

Given the image I, its edge map I_e and its supporting vector field $\boldsymbol{W}$, we define an 2D accumulator array $C_d(\cdot)$ for the directed Hough transform as:

$$C_d(\rho_k, \theta_l) = \sum_{\substack{p: \ |\angle(\boldsymbol{W}(p), \boldsymbol{n_l})| < \pi/2 \\ p_x \cos \theta_l + p_y \sin \theta_l - \rho_k = 0}} I_e(p) \tag{5}$$

where $\boldsymbol{n_l} = (\cos \theta_l, \sin \theta_l)$.

Our goal is to guarantee that each directed line l passing through pixels on the boundary of the ROI, such that the ROI is in its positive half-plane $H^+(l)$, has a correspondent local maxima in C_d. Hence we need to define a proper supporting vector field and a good edge map.

With the observation that, along the boundary of the ROI in a X-ray image, the directions of intensity gradient vectors are often from the shadow region (non-ROI) to the ROI, we define the supporting vector field (U, V) with the estimated gradient vector $\nabla I = (I_x, I_y)$. Special values are set in the four sides of the image. As in some cases, parts of the boundary of the ROI are on the sides of the image and there is no shadow region near it.

$$\boldsymbol{W}(p) = \begin{cases} (1,0) & \text{if } p_x = 1 \\ (-1,0) & \text{if } p_x = M \\ (0,1) & \text{if } p_y = 1,\ 1 < p_x < M \\ (0,-1) & \text{if } p_y = N,\ 1 < p_x < M \\ (I_x(p), I_y(p)) & \text{otherwise} \end{cases} \qquad (6)$$

where, $M \times N$ is the size of the image I. And we estimate the edge map I_e with a learning-based method to take advantage of the training data (see Sect. 2 for details). The complete directed Hough transform algorithm for the ROI boundary is shown in Algorithm 1.

Algorithm 1. Directed Hough Transform

Input: Image I
 1: Compute the supporting vector field $\boldsymbol{W}$. (Eq. 6)
 2: Compute the edge map I_e. (Sect. 2)
 3: **for** each pixel $p \in I$ **do**
 4:     **if** $I_e(p) > 0$ **then**
 5:         **for** $\theta \in [0, 2\pi)$ **do**
 6:             set $\boldsymbol{n} = (\cos\theta, \sin\theta)$.
 7:             set $\rho = p_x \cos\theta + p_y \sin\theta$.
 8:             **if** $|\angle(\boldsymbol{n}, \boldsymbol{W}(p))| < \pi/2$ **then**
 9:                 $C_d(\rho, \theta) = C_d(\rho, \theta) + I_e(p)$
10:             **end if**
11:         **end for**
12:     **end if**
13: **end for**

---

4 Optimal Quadrilateral Detection

To detect the optimal quadrilateral, we need to find the optimal group of four directed straight lines cropping the ROI.

Firstly, we detect a list of directed lines l_i, denoted as $L_{list} = \{l_i,\ i = 1 \ldots k\}$, using the directed Hough transform. l_i is in correspondence with local maxima (ρ_i, θ_i) in $C_d(\cdot)$, and $C_d(\rho_i, \theta_i) > \tau_1$, where τ_1 is a positive threshold. $\theta_i \leq \theta_j$, if $i < j$. As the optimal quadrilateral can also be regarded as the intersection of the positive half-plane of the four lines cropping it, l_i should also guarantee that its positive half-plane $H^+(l)$ contains the ROI. Let $R_{\tau_2} = \cup_{Pr(P) > \tau_2}(P)$ be a region that pixels in it have high probability in the ROI, where τ_2 is set to 0.96 in this study. Hence, each directed line l_i in L_{list} satisfies: $\frac{|R_{tau_2} \cap H^+(l_1)|}{|R_{tau_2}|} > \tau_3$, where $0 < \tau_3 < 1$.

Secondly, we detect a group of four directed lines $\boldsymbol{l_i} = (l_{i_1}, l_{i_2}, l_{i_3}, l_{i_4})$ in L_{list}, such that $i_1 < i_2 < i_3 < i_4$, satisfying that $\left|\theta_{i_{j+1}} - \theta_{i_j} - \pi/2\right| < \tau_\theta,\ j = 1, 2, 3,$

$|\theta_{i_1} - \theta_{i_4} + 3\pi/2| < \tau_\theta$, $|\theta_{i_{j+2}} - \theta_{i_j} - \pi| < \tau_\theta$, $j = 1, 2$, $\rho_{i_{j+2}} + \rho_{i_j} < -\tau_\rho$, $j = 1, 2$, to maximize the function f:

$$f_{l_i} = \beta \sum_{j=1}^{4} C_d(\rho_{i_j}\theta_{i_j}) - \sum_{j=1}^{3}(\theta_{i_{j+1}} - \theta_{i_j} - \pi/2)^2 - (\theta_{i_1} - \theta_{i_4} + 3\pi/2)^2 \qquad (7)$$

where, τ_θ, τ_ρ and β are positive constant parameters.

We optimize Eq. 7 with a constrained exhaustive search. For $l_i \in L_{list}$, let $s(i) = \min\{j|\theta_j > \theta_i + \pi/2 - \tau_\theta\}$, $e(i) = \max\{j|\theta_j < \theta_i + \pi/2 + \tau_\theta\}$, it is easy to see that, in the optimal group of $(l_{i_1}, l_{i_2}, l_{i_3}, l_{i_4})$, $s(i_j) \leq i_{j+1} \leq e(i_j)$, for $j = 1, 2, 3$. We compute $s(\cdot)$ and $e(\cdot)$ beforehand to speed up the exhaustive search.

With the optimal group of $(l_{i_1}, l_{i_2}, l_{i_3}, l_{i_4})$, we can generate the optimal quadrilateral as the detected region of the interest (ROI), as in Fig. 1c.

Table 1. Comparative success rate: our method (s^*), proposed directed Hough transform with gradient magnitude map (s^1), undirected Hough transform with proposed learning-based edge map (s^2), and the method in Mao et al. [5] (s^A). D_1' and D_2' are the sets of testing images in D_1 and D_2, respectively. The success rate is computed with expert identification. The running time of the method in [5] is about 15 s per case in average, and ours is about 1.8 s.

Success rate	s^*	s^1	s^2	s^A	Number of images
D_1'	**95 %**	90 %	82 %	80 %	1598
D_2'	**99 %**	96 %	94 %	85 %	705

5 Experiments

To show the robustness of our proposed method and the impact of the two components, we evaluate our algorithm on two data sets, D_1 and D_2, acquired by X-ray machines. We randomly select 100 images from the union of the two data sets for training, and evaluate our method on the remaining images. We also evaluate another learning based method [5] with the same training and testing images for comparison, in Table 1. D_1 and D_2 are from different sites and the difference of the success rate between D_1' and D_2' is due to larger variability of orientation and size of images in D_1.

For each image, our algorithm takes 1.807 s in average to detect the ROI, on an Intel Core i5-3470, 3.2 GHz processor and 4 GB memory system. The average size of input images is 350×350. The training of the selected images takes about 2 h, in which the training data are labeled as ROI or not in pixel-level. In Fig. 3, we show some results of our method, including a failure case Fig. 3e, in which, almost all the right side of the ROI is very weak.
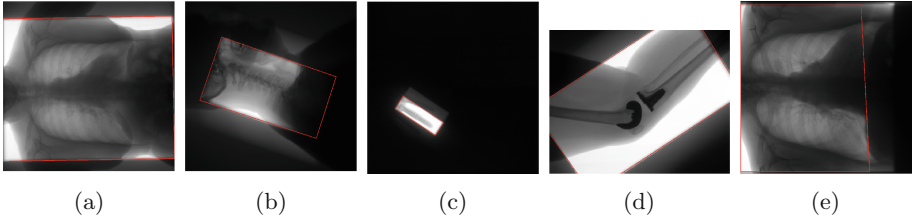
(a)   (b)   (c)   (d)   (e)

Fig. 3. Some results of our methods, (e) is a failure case

6   Discussion

In this paper, we propose a novel automatic algorithm to detect the region of interest in a X-ray image quickly and accurately. Learning-based edge maps are much more accurate than unsupervised methods, and the directed Hough transform add strict shape constraints for the quadrilateral detection. However, although we use a learning-based edge estimation, our method still cannot work if parts of the boundary are too weak, which makes the basic assumption of our method no longer hold.

References

1. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
2. Luo, J., Senn, R.A.: Collimation detection for digital radiography. In: SPIE (1997)
3. Sheth, R., et al.: Region of interest identification in collimated x-ray images utilizing nonlinear preprocessing and the radon transform. J. Electron. Imaging **14**(3), 033011 (2005)
4. Kawashita, I., et al.: Collimation detection in digital radiographs using plane detection hough transform. In: Palade, V., Howlett, R.J., Jain, L. (eds.) KES 2003. LNCS, vol. 2774. Springer, Heidelberg (2003)
5. Mao, H., et al.: Multi-view learning based robust collimation detection in digital radiographs. In: SPIE Medical Imaging (2014)
6. Achanta, R., et al.: SLIC superpixels compared to state-of-the-art superpixel methods. IEEE TPAMI **34**(11), 2274–2282 (2012)
7. Andrews, S., Hamarneh, G.: The generalized log-ratio transformation: learning shape and adjacency priors for simultaneous thigh muscle segmentation. IEEE TMI **34**(9), 1773–1787 (2015)
8. Canny, F.J.: A computational approach to edge detection. IEEE PAMI **8**(6), 679–698 (1986)
9. Duda, R.O., Hart, P.E.: Use of the hough transformation to detect lines and curves in pictures. Commun. ACM **15**(1), 11–15 (1972)