Anatomical Landmark Detection Using Nearest Neighbor Matching and Submodular Optimization

David Liu and S. Kevin Zhou

Siemens Corporation, Corporate Research and Technology, Princeton NJ, USA

Abstract. We present a two-stage method for effective and efficient detection of one or multiple anatomical landmarks in an arbitrary 3D volume. The first stage of nearest neighbor matching is to roughly estimate the landmark locations. It searches out of 100,000 volumes for the closest to an input volume and then transfers landmark annotations to the input. The second stage of submodular optimization is to refine the landmark locations by running discriminative landmark detectors within the search ranges constrained by the first stage results. Further it coordinates multiple detectors with a search strategy optimized on the fly to reduce the overall computation cost arising in a submodular formulation. We validate the accuracy, speed and robustness of our approach by detecting body regions and landmarks in a dataset of 2,500 CT scans.

1 Introduction

In the paper, we define an anatomical landmark (or landmark in brief) as a distinct point in a body scan that coincides with anatomical structures, such as liver top, lung top, aortic arch, iliac artery bifurcation, femur head left and right, to name but a few. Landmark detection is crucial for medical image applications. As a body region can be defined by landmark(s)[1], body region detection can be solved by landmark detection. Landmarks also provide seed points to initiate image segmentation [1] and registration[2]. In seminar reporting, the detected organ landmarks can help config the optimal intensity window for display [3] and offer the text tooltips for structures in the scan [4].

A practical landmark detection method must meet the following requirements. First, it must be robust to deal with pathological or anomalous anatomies such as fluid-filled lungs, air-filled colons, inhomogeneous livers caused by different metastasis, and resected livers after surgical interventions, different contrast agent phases, scans of full or partial body regions, extremely narrow field of views, etc. Figure 1 shows some examples of CT scans that illustrate the challenges. Second, since landmark detection is mostly a pre-processing step for computationally heavier tasks such as CAD and registration, it must run fast so that more time can be allocated for the heavier tasks. Finally, the landmark detection accuracy depends on the subsequent applications. For example, for body region detection exact 3D point positions are not needed; for registration, accurate landmarks are desired.

[1] A simple approach for determining the body region could rely on certain DICOM tags. But these tags are not always reliable, justifying a need for a dedicated image-based algorithm.

N. Ayache et al. (Eds.): MICCAI 2012, Part III, LNCS 7512, pp. 393–401, 2012.

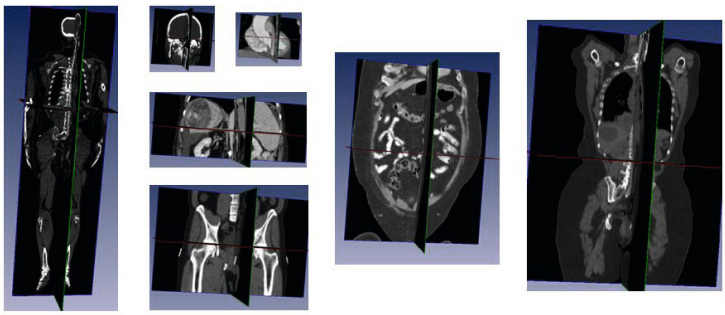


Fig. 1. Our dataset has 2,500 3D CT scans with different body regions and severe pathologies

We leverage the unitary, pairwise, and holistic contextual information (defined in Section 2) manifested in medical images to design an effective and efficient method for detection of one or multiple landmarks in a 3D volume. The proposed algorithm in Section 3 has two stages. First, nearest neighbor (NN) matching is to roughly estimate the landmark locations. It searches out of 100,000 volumes for the NN to the input query volume and then transfers the landmark annotations to the query. Second, submodular optimization is to refine the landmark locations by running discriminative landmark detectors within the search ranges constrained by the results from the first stage. Further it coordinates multiple detectors with a search strategy optimally determined on the fly to further reduce the overall computational cost arising in a submodular formulation. We validate the robustness, speed and accuracy of our approach by detecting one or multiple landmarks in a dataset of 2,500 CT volumes in Section 4.

2 Related Work and Context Exploitation

Designing a useful landmark detection method should effectively exploit the rich contextual information manifested in the body scans, which can be generally categorized as *unitary*, *pairwise or higher-order*, and *holistic* context. The *unitary context* refers to the local regularity surrounding a single landmark. The classical object detection approach in [5,6] exploits the unitary context to learn a series of supervised classifier to separate the positive object (herein landmark) from negative background. The complexity of this approach depends on the volume size. The *pairwise or higher-order context* refers to the joint regularities between two landmarks or among multiple landmarks. Liu et al. [7] embed the pairwise spatial contexts among all landmarks into a submodular formulation that minimizes the combined search range for detecting multiple landmarks. Here the landmark detector is still learned by exploiting the unitary context. In [8], the pairwise spatial context is used to compute the information gain that guides an active scheduling scheme for detecting multiple landmarks. Seifert et al. [9] encoded pairwise spatial contexts into a discriminative anatomical network. The *holistic context* goes beyond the relationships among a cohort of landmarks and refers to the whole relationship between all voxels and the landmarks; in other words, regarding the image as a whole. In [10], shape regression machine is proposed to learn a boosting regression function to

predict the object bounding box from the image appearance bounded in an arbitrarily located box and another regression function to predict the object shape. Pauly et al. [3] simultaneously regress out the locations and sizes of multiple organs with confidence scores using a learned Random Forest regressor. To some extent, image registration [11] can be regarded as using the holistic context too.

The proposed approach leverages all three contexts. The first stage of nearest neighbor (NN) matching exploits the holistic context. But instead of learning a regression function to capture the relationship between all voxels and the landmarks, we directly perform a full match and transfer the landmark annotations. This way we avoid scanning the image. The second stage of submodular optimization builds upon the approach in [7] that exploits both the unitary and pairwise contexts, but minimizes the overall computation instead of the total search range in [7]. Also, no holistic context is used in [7]. Instead, an 'anchor landmark' is first detected before triggering the whole detection process, utilizing only the unitary context in an exhaustive scanning.

3 Landmark Detection

3.1 Stage 1: NN Matching for Coarse Detection

Assume that a volume is represented by a D-dimensional feature vector. Given a query (unseen input) vector $x \in R^D$, the problem is to find the element y^* in a finite set Y of vectors to minimize the distance to the query vector:

$$y^* = \arg\min_{y \in Y} d(x, y) \tag{1}$$

where $d(.,.)$ is the Euclidean distance function. Other choices can be used too.

Volume Features. To facilitate the matching, we represent each volume by a D-dimensional feature vector. In particular, we adopt a representation of the image using 'global features' that provide a holistic description as in [12], where a 2D image is divided into 4×4 regions, eight oriented Gabor filters are applied over four different scales, and the average filter energy in each region is used as a feature, yielding in total 512 features. For our 3D volumes, we compute such features from nine 2D images, consisting of the sagittal, axial, and coronal planes that pass through 25%, 50%, and 75% of the respective volume dimension, resulting a 4,608-dimensional feature vector.

Efficient NN Search. In practice, finding the closest volume through evaluating the exact distances is too expensive when the database size is large and the data dimensionality is high. Two efficient approximations are used for speedup. Vector quantization is used to address the database size issue and product quantization [13] for the data dimensionality issue.

A quantizer is a function $q(.)$ mapping a D-dimensional vector x to a vector $q(x) \in Q = \{q_i, i = 1, 2, ..., K\}$. The finite set Q is called the codebook, which consists of K centroids. The set of vectors mapped to the same centroid forms a Voronoi cell, defined as $V_i = \{x \in R^D | q(x) = q_i\}$. The K Voronoi cells partition the space of R^D. The quality of a quantizer is often measured by the mean squared error between an input vector and its representative centroid $q(x)$. We use the K-means algorithm to find a

near-optimal codebook. During the search stage, which has high speed requirement, distance evaluation between the query and a database vector consists of computing the distance between the query vector and the nearest centroid of the database vector.

Our volume feature vectors are high dimensional (we use $D = 4608$ dimensions), which poses difficulty for a straightforward implementation of the K-means quantization described above. A quantizer that uses only $1/3$ bits per dimension already has 2^{1536} centroids. Such a large number of centroids makes it impossible to run the K-means algorithm in practice. Product quantization [13] addresses this issue by splitting the high-dimensional feature vector into m distinct sub-vectors as follows,

$$\underbrace{x_1, ..., x_{D^*}}_{u_1(x)}, ..., \underbrace{x_{D-D^*+1}, ..., x_D}_{u_m(x)} \tag{2}$$

The quantization is subsequently performed on the m sub-vectors $q^1(u_1(x)), ..., q^m(u_m(x))$, where $q^i, i = 1, ..., m$ denote m different quantizers. In the special case where $m = D$, product quantization is equivalent to scalar quantization, which has the lowest memory requirement but does not capture any correlation across feature dimensions. In the extreme case where $m = 1$, product quantization is equivalent to traditional quantization, which fully captures the correlation among different features but has the highest (and practically impossible, as explained earlier) memory requirement. We use $m = 1536$ and $K = 4$ (2 bits per quantizer).

Transferring Landmark Annotations. Given a query, we use the aforementioned method to find the most similar database volume. Assume this database volume consists of N landmarks with positions $\{s_1, ..., s_N\}$. We simply 'transfer' these landmark positions to the query. In other words, the coarsely detected landmark positions are set as $\{s_1, ..., s_N\}$. In the next section, we discuss how to refine these positions.

3.2 Stage 2: Submodular Optimization for Refined Detection

After the stage of NN matching, certain landmarks are located roughly. We now trigger the landmark detectors to search for a more precise position for each landmark only within local search ranges predicted from the first stage results. Running a landmark detector locally instead of over the whole volume reduces the computation and also reduces false positive detections. The local search range of each detector is obtained offline based on spatial statistics that capture the relative position of each pair of landmarks. Note that the two sets of landmarks in two stages can be different.

In order to speed up the detection, the order of triggering the landmark detectors needs to be considered. This is because, once a landmark position is refined by a detector, we can further reduce the local search ranges for the other landmarks by using the pairwise spatial statistics that embody the *pairwise context*. Consider a volume with N landmarks. Denote by $\Lambda_{(1):(n)} = \{l_{(1)} \prec l_{(2)} \prec ... \prec l_{(n)}\}, n \leq N$ the ordered set of landmarks that have been refined by detectors. Denote by U the un-ordered set of landmarks that remains to be refined. For each landmark $l_i \in U$, we define its search range $\Omega[l_i|\Lambda_{(1):(n)}]$ as the intersection of the search ranges predicted by the already detected landmarks:

$$\Omega[l_i|\Lambda_{(1):(n)}] = \bigcap_{j, l_j \in \Lambda_{(1):(n)}} \Omega[l_i|\{l_j\}], \tag{3}$$

where $\Omega[l_i|\{l_j\}]$ denotes the local search neighborhood for landmark l_i conditioned on the position of a detected landmark l_j.

Denote the volume of search range $\Omega[l_j|\Lambda]$ as $V(\Omega[l_j|\Lambda])$. Without loss of generality, assume the search volume is the cardinality of the set of voxels that fall within the search range. Denote by $\alpha[l_j]$ the unit computation cost for evaluating the detector for landmark l_j . Our goal is then to find the ordered set $\Lambda_{(1):(N)}$ that minimizes the *total computation*, i.e. ,

$$\Lambda'_{(1):(N)} = \underset{\Lambda_{(1):(N)}}{\operatorname{argmin}}\{\alpha[l_{(1)}]\ V(\Omega[l_{(1)}]) + \sum\nolimits_{i=2}^{N} \alpha[l_{(i)}]V(\Omega[l_{(i)}|\Lambda_{(1):(i-1)}])\}. \quad (4)$$

In [7], $\alpha[l_j] = 1$ for all j. This reduces to searching the minimum overall search range. We find that unit computation cost is roughly proportional to the physical disk size needed to store the detector model; hence we set $\alpha[l_(i)]$ as the model disk size.

As in [7], we use a *greedy algorithm* for finding the ordering $\{l_{(1)}, ..., l_{(N)}\}$ that attempts to minimize the overall cost proceeds as follows:

Initialize $\Lambda = \phi$.
for *j=1,...,N* **do**
  $l_{(j)} = \arg\min_k \alpha[k]V(\Omega[k|\Lambda])$;
  Append $l_{(j)}$ to the ordered set Λ so that the new $\Lambda = l_{(1)}, ..., l_{(j)}$.
end

In other words, in each round one triggers the detector that yields the smallest computation.

It is easy to prove that the overall cost function in Eq.(4) can be reducible to a submodular function [7]. Optimizing submodular functions is in general NP-hard [14]. One must in principle evaluate $N!$ detector ordering patterns. Yet amazingly, *the greedy algorithm is guaranteed to find an ordered set Λ such that the invoked cost is at least 63% of its optimal value [7]!* It is worth emphasizing that the ordering found by the algorithm is data-dependent and determined on the fly.

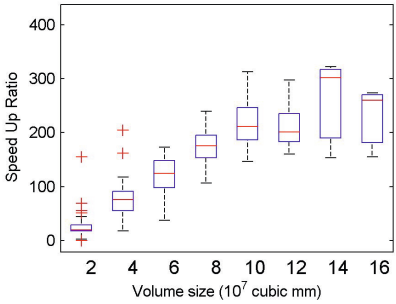
4 Experimental Results

We present two sets of experimental results. The first is on NN matching for body region detection and the second about fast and accurate detection of one or multiple landmarks. The system runs on an Intel Xeon 2.33GHz CPU with 3GB RAM.

4.1 NN Matching for Body Region Detection

In this experiment, we use the NN matching for detecting body regions that need rough landmark locations by utilizing the holistic context. Our matching-based approach requires a database sufficiently large so that, given a query, the best match in the training database indeed covers the same body region(s) as the query. We collect 2,500 volumes annotated with 60 anatomical landmarks, including the left/right lung tops, aortic arch, femur heads, liver top, liver center, coccyx tip, etc. We use 500 volumes for constructing the training database and the remaining 2,000 volumes for testing. To ensure that

each query finds a good match, we construct our database of 100,000 volumes in a near-exhaustive manner: In each iteration, we randomly pick one of the 500 volumes and then randomly crop and slightly rotate it into a new volume before adding it to the database. The annotated anatomical landmark positions in the original volume are transformed accordingly.



	Median time (ms)	Average of Median errors (mm)
Baseline [7]	450	28.6
Our method	5	29.9

Fig. 2. The performance of detecting body regions using NN matching. The left plot shows the speed up ratio vs. the volume size.

Registration based methods are not applicable since the test volumes cover a large variety of body regions. If each region is detected separately say using [6], the total detection time is proportional to the number of regions, as detecting each region requires a scan over the whole volume. The work in [15] reports a detection time around 2000 ms for 9 landmarks, and median distance error around 22mm on a GPU (parallelized) implementation. The work in [7] has the highest accuracy and fastest speed, so we compare against this work in better detail. As in Fig. 2, our implementation of [7], which is tuned to a similar detection accuracy as shown in Table 1, has a detection time of 450ms for 6 landmarks that define the presence of right lung, skull, aorta, sternum, and liver; but the maximum time is 4.9sec, significantly larger than the median. This poses a problem for time critical subsequent tasks. The proposed method has a nearly constant detection time of 5 ms, achieving a speed-up of 90 times while maintaining similar detection accuracy. The speed-up is even more significant if more regions are of interest as our detection does not depend on the number of regions. Our NN matching code can be optimized and parallelized for faster speed. In general, a large detection error from NN matching, which is fine for body region detection purpose, is due to the large variability in the landmark appearance and its relative location to other landmarks.

Table 1. Median detection errors (mm) for 6 different landmarks that define 5 body regions

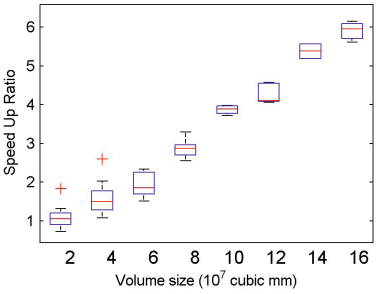
	Lung apex right	Skull base	Aortic root	Lung center	Sternum bottom	Liver center
Baseline [7]	24.1	31.9	23.2	20.6	37.3	35.2
Our method	27.1	19.1	35.8	24.5	35.1	37.9

4.2 Landmark Detection

When accurate positions are desired, we combine the NN matching with landmark detectors that exploit unitary context. Now each landmark detector only needs to search

in a local neighborhood around the rough position estimate given by the first stage, instead of searching in the whole volume. For detection of multiple landmarks, we further utilize pairwise spatial context for more improvements.

Detecting One Landmark. We consider detecting the liver top. In Fig. 3, the baseline approach uses a Probabilistic Boosting Tree (PBT) [6] to scan through the whole volume. Our method uses the PBT only to search in a local neighborhood. Evidently our method is much faster than the state-of-the-art due to the additional leverage of holistic context. A bigger volume yields more pronounced speedup (as large as 6-fold) as the use of holistic context breaks down the dependency on volume size.



	Median time (ms)	Average of Median errors (mm)
Baseline [6]	340	1.3
Our method	165	1.3

Fig. 3. The performance of accurately detecting the liver top

Detecting Multiple Landmarks. We further experiment accurately detecting 7 landmarks listed in Table 2 with three example landmarks of trachea bifurcation, liver bottom, and left kidney center shown in Fig. 4. Table 2 presents the mean detection error and the 95^{th} percentile error that exhibits the robustness of the combined approach. The results in [7] are also included for comparison. We obtain better detection results except for the left kidney center, whose annotations are quite ambiguous, while consuming less time with a mean computation of 1.1s vs 1.3s for [7]. Due to space limitation, we omit the results of 16 other organs and anatomical structures.

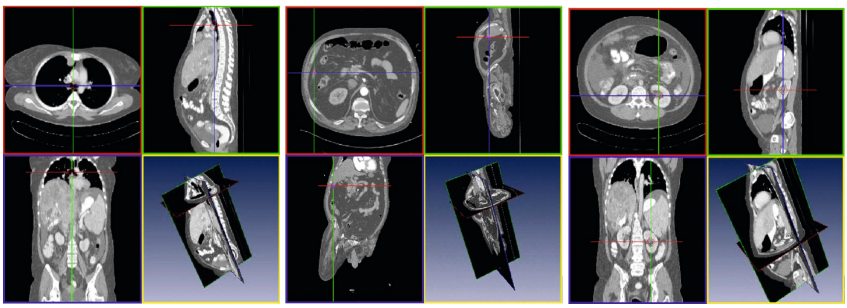


Fig. 4. Detected positions of trachea bifurcation, liver bottom, and left kidney center

Table 2. Errors (mm) in accurately detecting 7 different landmarks using NN matching and submodular optimization

(mm)	Mean	Q95	Mean [7]
Trachea bifurcation	**2.5**	4.5	2.8
Left Lung Top	**2.6**	6.0	3.2
Right Lung Top	**3.2**	8.5	3.7

(mm)	Mean	Q95	Mean [7]
Liver Top	**2.5**	4.0	2.9
Liver Bottom	6.4	30.5	n.a.
Left Kidney Center	8.4	50.7	**6.3**
Right Kidney Center	**6.4**	39.2	7.0

5 Conclusions

In this work we have introduced a fast and accurate method to detect landmarks in 3D CT data. Our method outperforms the state-of-the-art methods in detection speed with improved and comparable accuracy. The improvements arise from the leverage of holistic contextual information in the medical data via the use of an approximate NN matching to quickly identify the most similar database volume and transfer its landmark positions and the exploitation of unitary and pairwise context via a submodular formulation that aims to minimize the total computation for detecting landmark(s) and renders itself tocs a computationally efficiently greedy algorithm. Our method has been successively validated on a database of 2,500 CT volumes. In future we will extend it to different modalities such as MRI.

References

1. Rangayyan, R., Banik, S., Rangayyan, R., Boag, G.: Landmarking and Segmentation of 3D CT Images. Morgan & Claypool Publishers (2009)
2. Crum, W.R., Phil, D., Hartkens, T., Hill, D.: Non-rigid image registration: theory and practice. British Journal of Radiology 77, 140–153 (2004)
3. Pauly, O., Glocker, B., Criminisi, A., Mateus, D., Möller, A.M., Nekolla, S., Navab, N.: Fast Multiple Organ Detection and Localization in Whole-Body MR Dixon Sequences. In: Fichtinger, G., Martel, A., Peters, T. (eds.) MICCAI 2011, Part III. LNCS, vol. 6893, pp. 239–247. Springer, Heidelberg (2011)
4. Seifert, S., Kelm, M., Moeller, M., Mukherjee, S., Cavallaro, A., Huber, M., Comaniciu, D.: Semantic annotation of medical images. In: SPIE Medical Imaging (2010)
5. Viola, P., Jones, M.: Robust real-time face detection. Intl. J. of Comp. Vis. 57, 137–154 (2004)
6. Tu, Z.: Probabilistic boosting-tree: Learning discriminative models for classification, recognition, and clustering. In: Proc. ICCV, pp. 1589–1596 (2005)
7. Liu, D., Zhou, S.K., Bernhardt, D., Comaniciu, D.: Search strategies for multiple landmark detection by submodular maximization. In: Proc. CVPR (2010)
8. Zhan, Y., Zhou, X.S., Peng, Z., Krishnan, A.: Active Scheduling of Organ Detection and Segmentation in Whole-Body Medical Images. In: Metaxas, D., Axel, L., Fichtinger, G., Székely, G. (eds.) MICCAI 2008, Part I. LNCS, vol. 5241, pp. 313–321. Springer, Heidelberg (2008)
9. Seifert, S., et al.: Hierarchical parsing and semantic navigation of full body CT data. In: SPIE Medical Imaging (2009)
10. Zhou, S.K.: Shape regression machine and efficient segmentation of left ventricle endocardium from 2D B-mode echocardiogram. Med. Image Anal. 14, 563–581 (2010)

11. Izard, C., Jedynak, B., Stark, C.E.L.: Spline-Based Probabilistic Model for Anatomical Landmark Detection. In: Larsen, R., Nielsen, M., Sporring, J. (eds.) MICCAI 2006. LNCS, vol. 4190, pp. 849–856. Springer, Heidelberg (2006)
12. Torralba, A.: Contextual priming for object detection. Intl. J. Comp. Vis. 53, 169–191 (2003)
13. Jegou, H., Douze, M., Schmid, C.: Product quantization for nearest neighbor search. IEEE Trans. on Pattern Analysis and Machine Intelligence 33, 117–128 (2011)
14. Lovasz, L.: Submodular Functions and Convexity, pp. 235–257. Springer (1983)
15. Criminisi, A., Shotton, J., Bucciarelli, S.: Decision forests with long-range spatial context for organ localization in CT volumes. In: MICCAI Wksp. on Prob. Models for MIA (2009)