Semantic Localization-driven Partial Image Retrieval in CT Series

A. Cavallaro; H.-P. Kriegel; M. Petri; M. Schubert

1 Imaging Science Institute, University Hospital Erlangen, Erlangen, Germany; 2 Institute for Informatics, Ludwig-Maximilians-Universität München, Munich, Germany

**Summary**

**Background:** Picture archiving and communication systems (PACS) contain very large amounts of computed tomography (CT) data. When querying a PACS for a particular series, the user is often not interested in the complete series but in a certain region of interest (ROI), described e.g. by an example view in another series or an anatomical concept.

**Objectives:** Restricting a retrieval query to such an ROI saves both loading time and navigational effort. In this paper, we propose an efficient method for defining and retrieving ROIs.

**Methods:** We employ interpolation and regression techniques for mapping the slices of a series to a newly generated standardized height atlas of the human body.

**Results:** Examinations of the accuracy and the saved input/output (I/O) costs of our new method on a repository of 1,360 CT series demonstrate the advantages of our system. Depending on the scope of the retrieval query, we can economize up to 99% of the total loading time.

**Conclusion:** Our proposed method for flexible, context-based, partial image retrieval enables the user to directly focus on the relevant portion of the image material and it targets the high potential of I/O cost reduction of a common PACS.

**Keywords**

Information storage and retrieval, medical imaging, body regions, semantics, context-based image retrieval

**1. Introduction**

In commercial PACS, querying CT series is currently restricted to retrieving complete series. Simultaneously loading multiple CT series from a centralized server seriously strains the capacities of the available resources and the patience of the waiting users. In many cases, it is not necessary to display the complete series. Some users are only interested in a certain organ whereas others may want to see a body region corresponding to a focused region in another series. All of them want to see their region of interest (ROI) within a minimum waiting time.

Limited bandwidth is also a problem for medical image analytics. Sofka et al. [1] lately introduced a fast solution for landmark detection, which minimizes communication costs by applying a pyramid of varying image resolutions. This strategy is well-suited for divisible machine learning problems and visual presentation [2]. However, seeing a coarse resolution image while waiting for the complete image to load has often no additional value for a radiologist.

An alternative approach is to include a method into the PACS that quickly presents the high-resolution ROI to the user during the loading process, while optionally loading the remaining image material in the background. In [3], Ström and Cosman propose to use compression methods with lossless ROIs for decreasing waiting times. If the presented image region does not completely contain the ROI no harm has been done, except that the user has to wait for the complete series to load. The ROI is usually defined by the image content and not by its explicit coordinates. Hence, it is necessary to employ content-based methods to locate the CT slices intersecting the ROI.

The most established solution to locate ROIs is to annotate CT images with visible anatomical concepts [4]. Thus, a query can be computed by retrieving all images being labeled with the particular concept. To automatically generate these labels, organ and landmark detectors are applied. However, these methods are working on particular areas of the body which might not be displayed in all CT series. Furthermore, organ detectors try to locate the exact spatial boundaries and thus, they usually require an unnecessary computational overhead.

This is an approach of context-based image retrieval and it is not to be confused with content-based image retrieval, which usually relates to a similarity ranking of complete images [5–7]. The actual goal of content-based image retrieval in medicine is a combination of the two approaches: when given a query image, the query system should be smart enough to automatically determine the query context and therefore the locations of potential resulting similar images. In [8], Deserno et al. name this problem the Semantic Gap and identify it as one of the major remaining issues in content-based image retrieval. Consequently, a lot of work has been invested in the area of image registration, the mapping of images into a standardized anatomical reference system [9, 10]. These approaches are usually specialized to a local body part and they are expensive to compute.
fore, classical image registration approaches are not suitable for our purpose. Deserno et al. call this problem the Indexing Gap [8].

Recently, Criminisi et al. [11] applied regression forests to directly locate bounding boxes around certain organs in CT series. Though this approach does not result in exact organ outlines, it was shown to be considerably faster than more complex detection methods. Further extensions of the method [12] showed a very high robustness for varying body proportions. The main application of these methods is called semantic localization and it aims at the efficient real-time annotation in large PACS. However, the approach still predicts the positions of a fixed set of concepts and it cannot be easily extended to queries being specified by marking a query region in another CT series.

In this paper, we present methods for answering ROI queries retrieving the relevant slices from a CT series with a minimal overhead of irrelevant slices. We propose to map CT series to a height atlas describing the relative positions of anatomical concepts. Our method focuses on the height axis because it is the predominantly-used navigation axis for CT series. Given that the series are sliced along the transversal view, our approach is also applicable to other imaging modalities like Magnetic Resonance Imaging.

Partial retrieval queries for a target series \( v_i \) are answered in three steps (Fig. 1): First, the system determines a query interval in the height atlas corresponding to the user-specified ROI. This interval can be defined by either naming a concept (I) or by manually providing an example ROI in another CT series \( v_e \) (II). Next, the server determines the part of the query series corresponding to the query interval. Finally, the system transfers the selected slices from the image repository to the client workstation.

Technically, our system uses interpolation functions for mapping landmarks and other height coordinates from a specific series’ space to the standardized height scale and back. Landmarks are anatomically-relevant points within the body of low spatial variance. If no landmarks are available as matching points, we artificially generate some by using instance-based regression as in [13, 14]. In order to query a series for a particular semantic concept, our approach provides a body atlas storing the height distribution for multiple organs and landmarks. To summarize, the contributions of this paper are:

i. A method for learning a body atlas as an anatomical model, containing distribution parameters for landmarks and organs on the \( z \)-axis.

ii. A framework for mapping concept-based and example-based ROI queries into a standardized height space.

iii. A system for answering ROI queries allowing incomplete landmark annotations.

2. Method

In the following section, we specify our query types and give an overview of the proposed query pipeline. In Section 2.2, we describe two methods for registering a query ROI into a standardized height space, followed by our algorithm for learning a body atlas from semantically annotated example series in Section 2.3. The results of our experimental evaluation are presented in Section 3.

2.1 Workflow Overview and Notation

A PACS in our setting contains multiple CT volumes \( v_i \), represented as stacks of \( z(i) \) 2D-images or slices. The standardized height space \( H = [h_{\min}, h_{\max}] \) is a real-valued range representing the extension of the human body in the \( z \)-axis. A mapping function \( h_1: \mathbb{N} \rightarrow H \) maps slices of volume \( v_e \) to a standardized height value \( h \in H \). Correspondingly, the reverse mapping function \( s_1: H \rightarrow \mathbb{N} \) maps a position \( h \) in the standardized height space to a slice number \( s \) in \( v_i \). The particular functions will be described in Section 2.2. As a link between the two height spaces, we define matching points as triples \( p_{ij} = (s_{ij}, h_i, w_j) \in \mathbb{N} \times H \times \mathbb{R} \). One point \( p_{ij} \in P_i \) associates a slice number \( s_{ij} \in \{0, \ldots, z(i)-1\} \) of volume \( v_i \) to a standardized height value \( h_i \) in \( H \) and a reliability weight \( w_j \).
Table 1 displays an overview of the most frequently used parameters including some additional notations that will be introduced in the following sections. Figure 1 illustrates the complete workflow of query processing for concept-based queries and example-based ROI queries where the user provides an example ROI.

In our system, a concept-based ROI query retrieves a consecutive sequence of CT slices showing an anatomical content specified by the named concept $c \in C$, e.g., an organ like the heart or the liver. $C$ is the set of all considered query concepts. A body atlas contains one probability distribution for each concept $c$. This allows us to specify a standardized query range $[h_{lb}, h_{ub}] \subseteq H$ as

$$\arg\max_{h \in H}[P_i(H \leq h) \leq \tau], \arg\min_{h \in H}[P_i(H > h) \leq \tau]$$

for a threshold quantile $\tau$ and a concept $c$. (In Section 3.1, we demonstrate a good trade-off between completeness and retrieval speed for $\tau = 0.05$.)

In example-based ROI queries, the queried ROI is described by the content of a user-set of consecutive slices $(s_{lb}, s_{ub}, \ldots, s_{ub}) \subseteq H$ for an example series $v_i$. The standardized query range $[h_{lb}, h_{ub}] \subseteq H$ can be determined using a mapping function $h_i(s)$ on the slices of the given example ROI.

With the query interval $[h_{lb}, h_{ub}]$ given, the server then applies $s_i(h)$ to determine the result set of corresponding CT slices $(s_{lb}, s_{ub}, \ldots, s_{ub}) \subseteq [0, \ldots, z(i) - 1]$ in an example series $v_i$. The standardized query range $[h_{lb}, h_{ub}] \subseteq H$ can be determined using a mapping function $h_i(s)$ on the slices of the given example ROI.

For mapping atlas positions in $H$ to slices in $\mathbb{N}$, we vary the above interpolation function to a reverse interpolation model $s_i(h)(p)$ (Fig. 3). The transformation inverts the spacing $\delta_i$ and swaps the mapping spaces $\mathbb{N}$ and $H$.

$$h_i(s_{lb}, s_{ub}) \in \mathbb{R}$$

Standardized height space

One height value in $H$

One slice number of volume $v_i$ in $[0, \ldots, z(i) - 1]$

Matching point of $v_i$ with reliability weight $w_j$

Mapping functions $\mathbb{N} \rightarrow H$ and $H \rightarrow \mathbb{N}$ using matching points $P_j$

Additionally, a relevance function $rel'$ now uses position differences in the atlas scale instead of slice differences like the relevance functions $rel$.

In order to answer ROI queries on a series $v_j$, function $s_j(h)(p)$ requires at least two matching points as $P_i$. For this publication, we use the landmark detector of [17], which detects up to 22 landmarks within the thorax region. This detector, however, does not return any or not enough matching points for CT series with a blurred image quality or limited anatomical ranges. Thus, we recommend using the instance-based regression technique of [14] for generating matching points if none are available. This approach maps a single slice $s_{lb}$ into a height scale via $k$-nearest neighbor regression using an image descriptor for $s_{lb}$. We already use this method in a partial retrieval algorithm detailed in [13], which at the time of writing did not yet make use of a semantic localization atlas.

2.2 Mapping Functions

Our interpolation approach maps a series $v_i$ to the height space $H$ and back based on a set of available matching points $P_i$. Inspired by the work of Haas et al. [15], we tested various non-linear interpolation approaches for dealing with varying body proportions and imprecise matching points. Function $h_i(s_{lb}, s_{ub})$ maps slice number $s_{lb}$ to a height value $h_i \subseteq H$. It is adapted to the series $v_i$ by the set of matching points $P_i$ and the slice spacing $\delta_i$, describing the thickness of a slice in the target space $H$ (Fig. 2), where $rel'(h, p)$ is a relevance term for the matching point $p$ with respect to the input slice $s_{lb}$. The simplest relevance function, $(1\text{NN}) rel'(h, p) = 1$, is equivalent to applying a weighted linear interpolation. In [13], we used the inverse absolute difference, $\text{(ABS)} rel'(h, p) = \min(1,|s_{lb} - h|^r)$ as a relevance term, as it showed to be more stable than the squared inverse difference $(\text{SQIR})$ used in [15]. Other, more complex relevance functions showed to be even less suitable for obtaining a reliable height interpolation. In accordance with [16], we define a radial basis function kernel (RBF) as

$$\text{rel}''(h, p) = \exp\left( -\frac{(s_{lb} - h)^2}{\sigma_p^2}\right)$$

being the standard deviation of a matching point $p$ over the training dataset.

For mapping atlas positions in $H$ to slices in $\mathbb{N}$, we vary the above interpolation function to a reverse interpolation model $s_i(h)(p)$ (Fig. 3). The transformation inverts the spacing $\delta_i$ and swaps the mapping spaces $\mathbb{N}$ and $H$.

$$h_i(s_{lb}, s_{ub}) \in \mathbb{R}$$

Standardized height space

One height value in $H$

One slice number of volume $v_i$ in $[0, \ldots, z(i) - 1]$

Matching point of $v_i$ with reliability weight $w_j$

Mapping functions $\mathbb{N} \rightarrow H$ and $H \rightarrow \mathbb{N}$ using matching points $P_j$
2.3 Height Atlas Definition

The accuracies of our partial retrieval queries are mainly dependent on the quality of the matching points $P_i$. When using landmark detection for defining matching points, we require a standardized height atlas \((h, w) \in H^{|C|} \times R^{|C|}\), mapping the set of available concepts $C$ to standardized heights and reliability weights. There are two principal ways to generate such an atlas: using the real-world landmark positions of a template or example series or by forming a consensus model from a set of manually-aligned series. Since these approaches are biased, we additionally propose a refinement algorithm, which improves an existing atlas by iterative adaptation to a larger dataset.

2.3.1 Model by Example (ExMP)

Our example series for using the first approach ExMP is a publicly available full body scan showing a 1.75 m tall woman. We average the output of the landmark detector over varying resolution scales as $h$ and derive the weights $w$ from the variation of the detected coordinates for each landmark.

2.3.2 Manually Aligned Examples (ALIGN)

As an alternative, we form a consensus model ALIGN using a set of 33 CT series (12 female; 21 male patients): all series are manually mapped into a standard $[0, 1]$ scale ($0 =$ feet, $1 =$ head) by defining the start and end positions of the series and linearly interpolating the positions in between. For 12 volumes, all landmarks were found, two vol-

---

**Input:** \(s \in N^{n \times |C|}\): Slice numbers for $n$ volumes and $|C|$ concepts, $M_{\text{seed}} \in H^{|C|} \times R^{|C|}$: seed atlas \((h, w)\), $h$: interpolation function

1. **function** \(\text{AdaptModel}(s, M_{\text{seed}}, h)\)
2. \(M_0 \leftarrow M_{\text{seed}}\)
3. \(c := \text{LossMin} (s, M_{\text{seed}}, h)\)
4. \(P_t \leftarrow P_{t-1}\) \(\triangleright \) Generate matching points
5. \(P_t \leftarrow P_{t-1}\) \(\triangleright \) Generate new model
6. \(h_c \leftarrow \text{median}(h_{i, c})\) \(\{s_{i, c}, h_{i, c}\}\)
7. \(w_{\text{ed}} \leftarrow 1/\left|\text{std}(h_{i, c})\right|\) \(\{s_{i, c}, h_{i, c}\}\)
8. \(w_{\text{oc}} \leftarrow \text{size}(i \in \{1, \ldots, n\} | h_{i, c} \text{ was found})\)
9. \(\bar{w} \leftarrow \text{normalized} \bar{w}_{\text{ed}} + 0.5 \cdot \text{normalized} \bar{w}_{\text{oc}}\) \(\triangleright \) Combine weights
10. \(M_t \leftarrow (h, w)\) \(\triangleright \) File the new model
11. \(c_{t} := \text{LossMin} (s, M_t, h)\) \(\triangleright \) Evaluate
12. **if** \(c_{t-1} < c_{t} \text{ min improvement} \) **then** \(\triangleright \) No improvement
13. **return** \(M_{t-1}\) \(\triangleright \) Previous model

**Output:** New height atlas adapted to the dataset $s$

---

**Fig. 4** Atlas refinement algorithm

**Fig. 5** The distributions of landmark and organ boundaries of [17] when mapped to the atlas scale via $h_s$, $h_{s, q}$ using abs interpolation and ExMP seeding. The bars correspond to the probability (density in $[0, 1]$) to observe a landmark or organ within a range of 0.225 cm.

---

[..] volumes only resulted in one detected landmark and on average, 16 of the 22 landmarks were detected (with a standard deviation of 7). The standardized height $h_c$ of a concept $c$ is defined as the median of the annotated height values, and the weight $w_c$ is formed from a combination of the inverse standard deviation and the number of series $c$ occurs in. This model is less biased than EXMP, but it can be subject to annotation imprecision.

2.3.3 Improving a Standardized Height Atlas

Both atlases can be further improved by using a refinement procedure (c.f. the algorithm in Fig. 4) starting with the initial atlas $M_{seed} = (h, w)$. The algorithm requires a set of slice annotations $(s)_{i,q} \in N \times |C|$ for the set of detectable concepts $C$ within a database of $n$ volumes which may contain missing values as not all volumes will show all concepts and there may be false negatives of the detector.

In iteration step $t$, the concept positions $s$ are combined with the current model $M_t$ to generate a set of model-induced matching points $\hat{P}_i = P_{i, M_{t-1}} = (s_i, h, w)$ for each training series $v_i$. We apply $h_i(s_{i,q})$ to determine a new height value in $H$ for each series $v_i$ and each concept $c$ by excluding the current concept $c$ from the set $\hat{P}_i$. Then, a new atlas model is generated in steps 4–10 in a similar way as in the ALIGN model. The quality of the new model is evaluated using a least squares error function, $\text{loss}_{MSE}(s, M_t, h)$, accumulating the squared difference of the predicted concept positions $h_i(s_{i,q})\beta_i(s_{i,q}, h, w)$ from the newly defined atlas position $\hat{h}_i$. If the new model is an improvement, the algorithm proceeds with the next iteration $t+1$, else it returns the previous model $M_{t-1}$.

Let us note that the $H$ coordinates of the matching points generated in step 4 also define the standardized anatomical distributions used for determining the retrieval range $[h_{low}, h_{high}]$ for a query by concept. Figure 5 displays these observed density distributions in $H$ for the 22 landmarks in a set of 1,360 CT series. It also shows nine additional organ boundaries; hence these distributions have natural spread. The organ detectors used here were provided by [17].

3. Results

In this paper, we use the landmark detector of [17]. Any other landmark detector or even a combination of several detectors could be used as well. Our experiments were conducted on a heterogeneous database of 1,360 CT series from the THESEUS MEDICO data repository provided by the university of Erlangen where at least 20 landmarks could be detected, containing $7 \times 10^5$ 2D images with a total size of 360 gigabytes.

The dataset is characterized in Fig. 6: it consists of a varying number of slices covering body ranges of only 40 centimeters (cm) up to one meter of the scanned patient. The slice spacing and slice thickness are at most 0.5 cm.
3.1 Runtime Evaluation

We now demonstrate the advantages with respect to retrieval times of our solution over common PACS. Our goal is partial retrieval of CT series. Therefore, we could not use a real PACS. Instead, we resorted to a remote file system with a custom-designed indexing structure. While mapping into the height atlas with available matching points takes only 3 ms (milliseconds), the detection of the 22 landmarks of [17] usually requires between 3 to 10 seconds. In our group’s 100 megabit Ethernet, it takes 110 ms to load one slice of 512 × 512 pixels. Thus, this time investment of 3 seconds pays off if at least 30 slices can be excluded from the series to be loaded. Since the expected number of slices in our repository is 548, it is very likely that in most partial retrieval queries at least 30 slices can be omitted.

In Figure 7, we see histograms over the pruned number of slices for our used database of 1,360 CT series with four predefined retrieval ranges. The retrieval range is the part of the series to be loaded in centimeters. For a retrieval range of 50 cm, we already have a chance of 71% that at least 30 slices can be pruned. For smaller retrieval ranges this probability is by far higher. A 10 cm retrieval range, for instance, has a 95% chance of pruning at least 100 slices. Since Figure 7 was created for the inhomogeneous dataset summarized in Figure 6, the overall distributions are divided into bulks of CT series with a lower number of slices and CT series with many slices.

The retrieval range can be tuned with the $\tau$ parameter, introduced in Section 2.1. We examine the effect of $\tau$ on the retrieval ranges of the average landmark, the average organ and the heart as a single organ in Figure 8. Our chosen setting of $\tau = 0.05$ implies a certainty of 90% that the requested range is actually loaded. Figure 8 shows that on the average, 2 cm need to be loaded for a landmark query and that 34 cm need to be loaded for an average organ query, respectively. The heart, which is more compact than, for instance, the lungs has a retrieval range of 15.5 cm. Taking into account the information of Figure 7, even the retrieval ranges of organs allow a clear runtime improvement.

Using the online matching point generating method of [13], both the data reduction and the time gain are even stronger, as one artificial matching point via regression can be generated in only 42 ms. If, for instance, only a certain transversal view is requested, the retrieval range of a volume of 1000 slices and more can quickly be narrowed down to only 10 slices, therefore resulting in a speed gain of up to 99%.

3.2 Retrieval Accuracy

For validating the quality of our atlas generation method, we evaluate all combinations of atlas seedings and interpolation functions. The atlas is built and validated within a 10-fold cross-validation setting on 1,360 CT series providing at least 20 landmarks, the same dataset used for the visualization in Figure 5. Consequently, no tested volume has any influence on its training set. The average localization errors of a single landmark...
Fig. 9 Leave-one-out localization errors (in cm) of \( h_t \) (series to atlas) and \( s_i \) (atlas to series positions) for the seeding approaches \texttt{EXMP} (model by example), \texttt{ALIGN} (manually aligned examples) and \texttt{RAND} (random) using 20 matching points with the relevance functions \texttt{ABS} (inverse absolute difference), \texttt{SQR} (inverse square difference), \texttt{RBF} (RBF kernel) and \texttt{LIN} (linear interpolation). The number of iterations required for the training algorithm is displayed at the bottom of the bars. The \texttt{RAND} experiments do not show any seed errors for the anatomically meaningless initial atlases. Since the \texttt{SQR} and \texttt{RBF} interpolation approaches do not converge to any interpretable atlas, they are not displayed as well. \texttt{SQR} interpolation even got caught in an endless loop, denoted by Inf at the position of the bars.

Fig. 10 Leave-one-out localization errors (in cm) for a variable number of used matching points (\texttt{EXMP} seeding) with the same notation as in Figure 7. The solid bars display the performance of \( h_t \) (series to atlas), the white dashed bars display the performance of \( s_i \) (atlas to series positions).

mark (predicted by using only the remaining landmarks as matching points) are displayed in Figure 9. We transformed the error values to the range of the initial \texttt{EXMP} mapping of 0–175 cm for a correct comparison of the different mappings.

Figure 9 shows that \texttt{ABS} and \texttt{LIN} interpolation are comparably successful with average errors around 1 cm, while the locally more adaptive mappings \texttt{SQR} and \texttt{RBF} (and other, more complex interpolation approaches not shown here) have a stronger susceptibility to outliers, and therefore generate worse mapping results. Atlases based on \texttt{EXMP} seedings converge to comparable errors like those based on \texttt{ALIGN} seedings. However, the \texttt{EXMP} seedings really profit more from the atlas refinement algorithm (cf. the original seed error in dashes). The original \texttt{ALIGN} atlas formed on a set of manually-aligned series is not significantly improved by using our refinement algorithm. When testing random seedings (\texttt{RAND}: uniformly sampled height positions and weights in \([0, 1]\)), atlases of comparable quality resulted for \texttt{ABS} and \texttt{LIN}, whereas \texttt{SQR} and \texttt{RBF} re-ordered the landmarks into disjoint sub-clusters due to local minima. The most surprising result of these experiments is the good performance of the simple weighted linear interpolation \texttt{LIN}. Let us note that for the pre-dominant spacing in our database of 0.5 cm, an average precision of 1 cm is only a deviation of two slices.

Figure 10 demonstrates the effect of the number of available matching points for \texttt{EXMP}-seeded models. Other seedings show the same trend: there is a strong decrease of the prediction errors when using five instead of two matching points. Using even more matching points results in further but smaller prediction improvements.

Finally, Figure 11 shows the stability of our retrieval approach by displaying extreme values and the standard deviation for \text{EXMP} interpolation (with the inverse absolute difference as relevance function) based on \text{EXMP} seeding. Note that this figure corresponds to the first set of bars in Figure 10. Once, the mapping function can use at least five matching points, even the maximum error stays below 20 cm for both the mapping from the series to \( H_t \), \( h_t \) and the reverse direction, \( s_i \). Five matching points are consequently a good compromise between the effort of generating matching points and a good localization.
Note that matching points generated by regression can be validated by their leave-one-out prediction error in a set of height-labeled CT series. In [13], we reported an error of 1.88 cm for a query time of 42 ms. Any further increase in accuracy comes at the price of longer query times due to more detailed slice descriptors (e.g. an error of 1.21 cm for 3 seconds per matching point).

Therefore, even though the precision of regression matching points is not as good as the precision based on landmark detection (with errors of 1 cm), using the regression approach allows queries on CT series with missing or error prone landmark annotations.

4. Conclusion

In this paper, we proposed an efficient method for processing partial image retrieval queries using a semantic localization approach on a repository of CT series with the goal of decreased I/O costs. By naming a retrieval context either verbally as anatomical location or by providing an example region, the requested region of interest is inferred from a general height atlas of the human body. This allows the focused retrieval of arbitrary height ranges within the query series even without requiring prior annotation.

Our system is based on interpolation functions using so-called matching points to link CT series to the height atlas. Additionally, the proposed system allows generating matching points based on imaging techniques if no landmarks are available. In our experimental evaluation, we demonstrate the accuracy and runtime costs of our approach.

We conclude that with low runtimes of our height mapping, we can drastically decrease the retrieval times of partial CTs compared to the retrieval of a complete series. A clinical retrieval system can thus save or prioritize valuable time as well as hardware resources by allowing the users to focus on volume portions which are actually relevant to the current anatomical context. For future work, we plan to extend our system to partial CT retrieval in all three dimensions and to test further landmark types.

Acknowledgments

This research has been supported in part by the THESEUS program in the MEDICO project. It is funded by the German Federal Ministry of Economics and Technology under the grant number 01MQ07020. The responsibility for this publication lies with the authors.

References