Segmentation of Multiple Knee Bones from CT for Orthopedic Knee Surgery Planning

Dijia Wu, Michal Sofka, Neil Birkbeck, and S. Kevin Zhou⋆

Imaging & Computer Vision, Siemens Corporate Technology, Princeton, NJ, USA

Abstract. Patient-specific orthopedic knee surgery planning requires precisely segmenting from 3D CT images multiple knee bones, namely femur, tibia, fibula, and patella, around the knee joint with severe pathologies. In this work, we propose a fully automated, highly precise, and computationally efficient segmentation approach for multiple bones. First, each bone is initially segmented using a model-based marginal space learning framework for pose estimation followed by non-rigid boundary deformation. To recover shape details, we then refine the bone segmentation using graph cut that incorporates the shape priors derived from the initial segmentation. Finally we remove overlap between neighboring bones using multi-layer graph partition. In experiments, we achieve simultaneous segmentation of femur, tibia, patella, and fibula with an overall accuracy of less than 1mm surface-to-surface error in less than 90s on hundreds of 3D CT scans with pathological knee joints.

1 Introduction

American Academy of Orthopedic Surgeons reports that over 500,000 patients have their degenerative knees replaced each year in United States. In recent years, the knee replacement procedure has advanced with personalized surgery designed specifically for each patient. The procedure starts with a CT scan of patient's knee joint from which a 3D knee anatomy model of this patient is extracted. This model is then used for subsequent knee surgery planning. To streamline workflow and reduce cost, fully automatic and highly accurate segmentation of knee bones from 3D CT images is critical in clinical practices.

Most of previous studies on automatic knee bone segmentation focused on MR data, including the voxel based [1] or block-wise classification [2] with texture features and intensity distribution. However, all these methods are ineffective in dealing with the strong intensity and texture inhomogeneities between cortical and cancellous bone in CT and MR images. To improve the segmentation robustness, statistical shape models [3] are often used as prior knowledge to guide the segmentation [4–6]. In these methods, fast and accurate model initialization and adaptation remains a challenge. Graph-based algorithms [7] have been extensively used to solve different vision problems, including bone segmentation [2, 4, 8–10] as well; but the accuracy of such algorithms usually depend on seed points often manually provided. Also the bones are segmented individually instead of jointly, which often leads to sub-optimal segmentation results that even

---
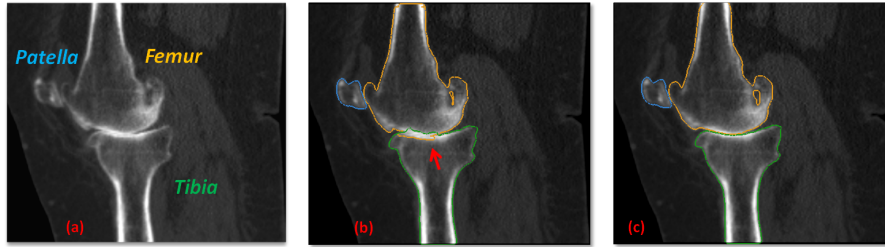
Fig. 1. (a) Example CT image of femur and tibia where two bones touch each other. (b) Segmentation result where overlap occurs. (c) Joint segmentation results.

overlap with each other particularly in regions where bones are too close or touch each other. This happens more often in the osteoarthritis patients with degenerative cartilage as shown in Fig.1(a). To handle the bone overlap, Li *et al.* [11] proposed a novel column graph-based algorithm to solve coupled surface segmentation problems, which was later used for simultaneous bone and cartilage segmentation in the knee [12]. It exploits the geometric constraints between multiple terrain-like and cylindrical surfaces; but unfolding the structures like femur with two condyles to terrain-like surfaces is nontrivial. Kainmueller *et al.* [13] proposed coupled deformable model for multiple-object segmentation, which does not completely prohibit but discourages the overlap.

Here we present an approach for segmenting multiple knee bones that makes two key contributions. The first contribution is its novel combination of three state-of-the-art methods for *precise* segmentation of multiple knee bones of diseased knees: (i) Marginal space learning(MSL) [14]. Each bone is first detected using MSL and then deformed with a statistical shape model [3]. (ii) Graph cut [7]. The adapted model is then used as a shape prior in a graph cut formulation for refined segmentation. (iii) Multi-layer graph cut [18]. Because each bone is separately segmented, their results possibly overlap. We utilize multi-layer graph cut to remove such overlap error. The second contribution is its full automation and computational efficiency because it needs no image unwrapping and the joint segmentation can be performed in a small local region where the overlap occurs. This efficiency is *clinically significant* for reduced cost and streamlined workflow. Currently our approach is already deployed in Siemens image-to-implant system.

2 Learning-based Bone Detection and Model Fitting

2.1 Pose Estimation

For a given volume I, each bone is first individually detected from the volume by searching for the optimal similarity transformation parameters or pose parameters including translation $\mathbf{t} = (t_x, t_y, t_z)$, orientation $\mathbf{r} = (r_x, r_y, r_z)$ and anisotropic scaling $\mathbf{s} = (s_x, s_y, s_z)$. The pose estimation task can be formulated by maximizing the posterior probability as follows:

$$(\hat{\mathbf{t}}, \hat{\mathbf{r}}, \hat{\mathbf{s}}) = \arg\max_{\mathbf{t}, \mathbf{s}, \mathbf{r}} P(\mathbf{t}, \mathbf{s}, \mathbf{r}|I). \tag{1}$$

Solving equation (1) involves the search in a nine dimensional parameter space, which can be computationally expensive in practice. Here we adopt an

efficient inference scheme, MSL[14], to decompose the whole search space into marginal space inference. The object localization is split into three steps: position estimation, position-orientation estimation, and full similarity transformation estimation.

$$(\hat{\mathbf{t}}, \hat{\mathbf{r}}, \hat{\mathbf{s}}) \approx (\arg\max_{\mathbf{t}} P(\mathbf{t}|I), \arg\max_{\mathbf{r}} P(\mathbf{r}|I, \hat{\mathbf{t}}), \arg\max_{\mathbf{s}} P(\mathbf{s}|I, \hat{\mathbf{t}}, \hat{\mathbf{r}})). \qquad (2)$$

After each step only a limited number of best candidates is kept to reduce the search space and speed up the inference. To learn the marginal posterior probabilities in Eq.(2), discriminative classifiers such as the probabilistic boosting tree (PBT) [15] or the probabilistic boosting network [16] can be used. Moreover, 3D Haar features are used for location detection and steerable features are used for orientation and scale inferences [14]. Fig.2(a) shows an example of the pose estimation result, where the pose parameters are represented as a bounding box.

2.2 Model Initialization and Boundary Deformation

After pose estimation, the shape of the target object is initialized using the statistical shape model (SSM) as follows:

$$\mathbf{x} = f(\mu; \hat{\mathbf{t}}, \hat{\mathbf{r}}, \hat{\mathbf{s}}), \qquad (3)$$

where $\mathbf{x}$ denotes the initialized shape, f is the rigid transformation with the pose parameters $(\hat{\mathbf{t}}, \hat{\mathbf{r}}, \hat{\mathbf{s}})$ estimated by MSL, μ represents the mean of the statistical shape model obtained from the training annotations.

The initialized shape is then deformed with the boundary detectors. Here, boundary detection is again formulated as a classification problem: whether there is a boundary passing point at (X, Y, Z) with orientation (O_x, O_y, O_z). The boundary detectors are used to move the mesh points on the current estimated shape surface along its normal direction to the optimal position, where the classification score is the highest. After adjustment, the deformed shape is projected to the SSM subspace to smooth out and constrain the surface. In our experiments, the dimension of the SSM subspace is selected to capture 98% of the shape variations from the training annotations. The process is repeated a few iterations until convergence. As an example shown in Fig.2(b), the derived shape after boundary deformation fits the image well, but is still subject to noticeable errors due to the loss of shape details by the statistical shape model, as well as possible boundary detection errors.
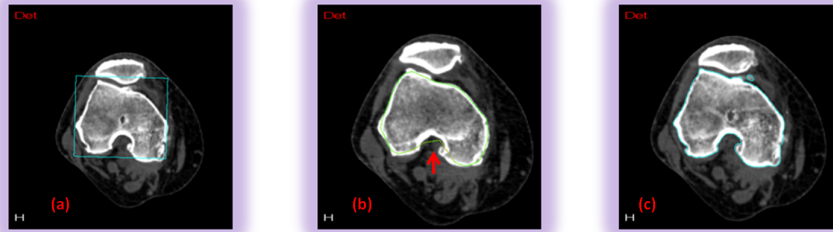
3 Bone Refinement with Shape Prior in Graph Cut

To further improve the accuracy, we formulate the following graph-based energy function with the previous segmentation result used as the shape prior:

$$E(L) = \sum_{p \in \mathcal{P}} D_p(L_p) + \sum_{(p,q) \in \mathcal{N}} V_{p,q}(L_p, L_q), \qquad (4)$$

where $L = \{L_p \mid p \in \mathcal{P}\}$ is the binary labeling ($L_p \in \{0,1\}$) of volume $\mathcal{P}$, $D_p(L_p)$ is the unary data term which is defined as below:

$$D_p(L_p) = L_p(1 - g(\mathbb{M}(p))) + (1 - L_p)g(\mathbb{M}(p)). \qquad (5)$$

Fig. 2. The example result of (a) MSL pose estimation, (b) boundary detector based deformation, and (c) graph cut based refinement.

Here, $\mathbb{M}(p)$ measures the signed shortest distance of voxel p to the boundary of the prior segmentation. $\mathbb{M}(p) > 0$ when p lies inside the segmentation (foreground), $\mathbb{M}(p) < 0$ if p is outside (background), and $\mathbb{M}(p) = 0$ if p locates on the boundary. Therefore, $\mathbb{M}$ can also be viewed as a confidence map of the prior segmentation. The larger (smaller) $\mathbb{M}(p)$ is, the more likely voxel p should be classified as the foreground (background). When voxel p approaches the boundary ($\mathbb{M}(p) \approx 0$), label L_p becomes more uncertain and therefore more likely to be updated by the graph cut refinement. $\mathbb{M}(p)$ can be efficiently computed in linear time using the convolution method [17]. The sigmoid function $g(.)$ is defined as $g(x) = \frac{1}{1+e^{-x/\tau}}$, where τ is the parameter that controls the range of uncertainty of the previous segmentation result. In Eq.(4), $\mathcal{N}$ is the set of all pairs of neighboring voxels and $V_{p,q}$ is the pairwise interaction term:

$$V_{p,q} = \lambda e^{-\frac{(I_p - I_q)^2}{2\sigma^2}} \delta(L_p \neq L_q), \tag{6}$$

where $\delta(.)$ is the Kronecker delta function $\delta(L_p \neq L_q) = 1$ if $L_p \neq L_q$ and equal to 0 otherwise, λ and σ are the regularization parameter and contrast coefficient, respectively, and I_p and I_q denote the intensity of voxels p and q. The pairwise term encourages the neighboring voxels with similar intensities to be assigned the same label.

The segmentation is refined by minimizing the energy in Eq.(4) using the max-flow/min-cut algorithm. Fig.2(c) shows the improved result after graph based refinement.

4 Joint Bone Segmentation

Because each bone is separately initialized and refined in the previous steps, the overlap error can not be prevented when two bones touch each other, as one example shown in Fig.1(b). To remedy this, we present a joint re-segmentation method to co-segment the pair of bones that overlap in the initial segmentation. With specific spatial exclusion constraint introduced [18], the joint segmentation can guarantee a complete removal of erroneously overlapping boundaries.

Without loss of generality, we denote the pair of bones as A and B. L_A and L_B stand for the labeling of bone A and B, respectively. It means that voxel p is inside bone A if $L_{A(p)} = 1$ and otherwise if $L_{A(p)} = 0$, likewise for bone B. The
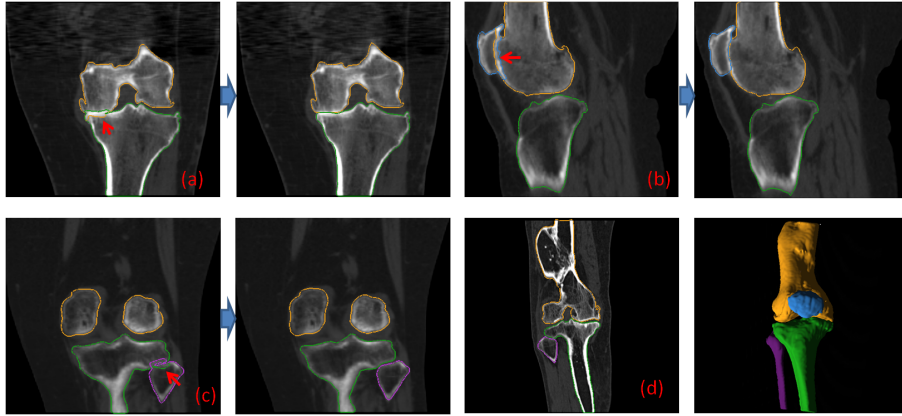
Fig. 3. The example of overlap error removed by joint segmentation. (a) pair of femur and tibia (b) pair of patella and femur (c) pair of tibia and fibula (d) pathological example with osteoporosis and 3D rendering of the segmentation.

energy function Eq.(4) can thus be extended to the case of two bones as follows:

$$E(L_A, L_B) = E(L_A) + E(L_B) = \sum_{p \in \mathcal{P}} D_{A(p)}(L_{A(p)}) + \sum_{(p,q) \in \mathcal{N}} V_{A(p,q)}(L_{A(p)}, L_{A(q)})$$
$$+ \sum_{p \in \mathcal{P}} D_{B(p)}(L_{B(p)}) + \sum_{(p,q) \in \mathcal{N}} V_{B(p,q)}(L_{B(p)}, L_{B(q)}), \tag{7}$$

where all the symbols follow exactly the same meaning as in Eq.(4) except that $\mathbb{M}$ is now based on the segmentation result after refinement as described in Section 3. As shown in Eq.(7), the minimization of $E(L_A, L_B)$ can be decomposed to the minimization of $E(L_A)$ and $E(L_B)$ separately because no interaction terms between L_A and L_B exist in the energy function Eq.(7). Therefore bone A and B are essentially segmented separately.

There is, however, a spatial exclusion constraint between L_A and L_B because bone A and B can not overlap in the space. It means that if $L_{A(p)} = 1$, $L_{B(p)}$ must $= 0$, and vice versa. This spatial constraint can be easily incorporated into the energy function Eq.(7) by adding the pairwise terms as follows:

$$\tilde{E}(L_A, L_B) = E(L_A, L_B) + \sum_{p \in \mathcal{P}} W(L_{A(p)}, L_{B(p)}), \tag{8}$$

where

$$W(L_{A(p)}, L_{B(p)}) = \begin{cases} +\infty & \text{if } L_{A(p)} = L_B(p) = 1 \\ 0 & \text{otherwise} \end{cases}$$

Therefore the optimal solution that minimizes the energy function $\tilde{E}(L_A, L_B)$ guarantees that $L_{A(p)}$ and $L_{B(p)}$ can not be both 1 at the same time ($\forall p \in \mathcal{P}$). However, the introduced pairwise term $W_{A(p),B(p)}(L_{A(p)}, L_{B(p)})$ is supermodular because $W(0,1) + W(1,0) < W(0,0) + W(1,1)$, hence it can not be directly

sym. surface error (mm)	mean	std. dev.	min	median	80 percentile
Femur (boundary deformation)	1.20	3.22	0.38	0.85	1.22
Femur (graph cut refinement)	0.83	3.35	0.43	0.55	0.67
Femur (joint segmentation)	0.82	3.33	0.43	0.55	0.67
Tibia (boundary deformation)	1.07	1.39	0.42	0.79	1.13
Tibia (graph cut refinement)	0.70	1.28	0.42	0.55	0.63
Tibia (joint segmentation)	0.69	1.25	0.42	0.55	0.63
Fibula (boundary deformation)	1.26	4.57	0.31	0.47	0.65
Fibula (graph cut refinement)	0.98	4.31	0.38	0.53	0.59
Fibula (joint segmentation)	0.96	4.29	0.38	0.53	0.59
Patella (boundary deformation)	0.72	2.07	0.35	0.62	0.67
Patella (graph cut refinement)	0.68	2.05	0.33	0.55	0.62
Patella (joint segmentation)	0.68	2.06	0.33	0.54	0.61

Table 1. The statistics of symmetric surface segmentation errors. All units are in mm.

optimized via min-cut/max-flow algorithm [19]. To address this problem, we flip the binary meaning of label $\bar{L}_B = 1 - L_B$, then the new energy function $\tilde{E}(L_A, \bar{L}_B)$ becomes submodular everywhere and min-cut/max-flow can be used to find the optimal labeling of L_A and L_B jointly.

The major advantage of the multi-layer graph cut is its 'simplicity'. The existing methods [10, 12, 13] based on the multi-column graph segmentation [11] all require point correspondence on adjacent surfaces for coupled deformation. Thus they are limited to terrain-like or cylindrical surfaces. Otherwise, complex methods are required to find so called shared profiles [13] or electric lines of force [10] which connect corresponding points on two surfaces and constitute the non-intersecting columns of the graph. By contrast, multi-layer graph cut makes no assumption of the shape of interactive surfaces, and does not need any such preprocessing. Hence it is easy to implement and also runs fast, taking less than 1s to remove all the segmentation overlaps in our implementation.

5 Experiments

In the experiment, we collect 465 CT volumes around the knee with an average size of $200{\times}200{\times}150$ voxels and $1{\times}1{\times}1mm$ voxel spacing, as shown in Fig.1. The annotations are obtained by experts based on visual assessment and consensus review. We randomly select 217 volumes for training the learning pipeline in Section 2, and use the remaining 248 volumes for testing. First, we compare the segmentation result after each step in the proposed method. As error measure we first computed the shortest Euclidean distances between each result mesh and its corresponding annotated mesh at every vertex of the former as well as every vertex of the latter, and then averaged all such distances. As shown in Table 1, the proposed method achieves quality segmentation with average symmetric surface error lower than $1mm$ for all four bones, and that is less than the 1mm voxel resolution. The graph cut refinement with shape prior decreases the mean error about $5 \sim 35\%$. Still we have failed cases (albeit very few) that contribute to the large variance in the surface error.

femur segmentation	1	2	3	4	5	6	7	8	9	10
before joint segmentation	3493	2075	1165	912	802	237	228	213	146	137
after joint segmentation	161	40	27	4	3	86	18	1	0	0
reduction %	95.4	98.1	97.7	99.6	99.6	63.7	92.1	99.5	100	100
tibia segmentation	1	2	3	4	5	6	7	8	9	10
before joint segmentation	3791	2729	1745	1562	896	522	491	448	312	286
after joint segmentation	1198	257	142	838	602	515	62	324	302	302
reduction %	68.4	90.6	91.9	46.4	32.8	1.3	87.4	27.7	3.2	-12.9

Table 2. The size (mm^3) of the overlap area between femur segmentation and tibia ground truth also between tibia segmentation and femur ground truth.

The overlaps after initial segmentation could happen when two bones almost touch each other (cartilages severely worn out). Those volumes are about 5% to 10% in our database. But this overlap usually occurs only around touching surfaces of neighboring bones, a very small region compared to the whole bone surface, so the numerical improvement by joint segmentation is not that obvious in Table 1. We use a different measurement to evaluate the effectiveness of the joint segmentation step to remove the overlap error. As shown in Table 2, we compare the size of the overlap area between *femur* result and *tibia* ground truth, as well as between *tibia* result and *femur* ground truth. For brevity, we only listed the 10 worst volumes with the largest overlap errors generated from previous individual segmentation step. The results show that the proposed joint segmentation can significantly reduce the overlap up to 100% in most of the cases, especially for the femur segmentation. Consistent improvements (except one case[1]) can be observed for the pair of patella and femur and the pair of tibia and fibula too. The joint segmentation result also depends on initial segmentation result of each bone. Fig. 3 shows examples of the overlap error eliminated by joint segmentation with a pathological example.

For each bone, the individual segmentation as described in Section 2 and 3 takes about 20 seconds to complete (Intel®Core^{TM}CPU @ 2.29 GHz and 3.23GB RAM). Because the joint re-segmentation in Section 4 is only applied to the local overlap region, it can be computed efficiently and only takes about 1 second on the average. If the the initial segmentation does not overlap, the joint re-segmentation can be skipped with little extra computational cost.

6 Conclusion

In this work, we present a fully automated method and system for segmenting multiple knee bones from 3D CT images. Our novel combination of marginal space learning, graph cut with shape-prior, and joint multi-layer graph cut for overlap removal achieves sub-mm segmentation accuracy needed for orthopedic surgery planning with a running speed of less than 90s for reduced cost and streamlined planning workflow in clinical practices, while guaranteeing no over-

---

[1] If initial femur segmentation leaks into tibia too much, it will affect the joint segmentation result of tibia adversely. Thats why joint Tibia #10 is worse, although it is very rare.

lap in the segmentation results between knee bones. In future, we plan to apply it for segmentation of other objects such as organs from other imaging modalities.

References

1. Bourgeat, P., at al.: MR image segmentation of the knee bone using phase information. *Med Image Anal*, **11**, 325–335, 2007.
2. Ababneh, S. Y., at al.: Automatic graph-cut based segmentation of bones from knee magnetic resonance images for osteoarthritis research. *Med Image Anal*, **15**, 2011.
3. Cootes, T. F., at al.: Active Shape Models - Their Training and Application. *Comput. Vis. Image Und.*, **61**, 1, 38-59, 1995.
4. Seim, H., et al.: Model-based auto-segmentation of knee bones and cartilage in MRI data. *Medical Image Analysis for the Clinic: A Grand Challenge, Beijing*, 2010
5. Schmid, J. and Magnenat-Thalmann, N.: MRI Bone Segmentation Using Deformable Models and Shape Priors. *Proc. MICCAI*, 2009.
6. Fripp, J., at al.: Automatic segmentation of the bone and extraction of the bone-cartilage interface from magnetic resonance images of the knee *Phys. Med. Biol.*, **52(6)**, 1617-1631, 2007.
7. Boykov, Y.: Graph cuts and efficient ND image segmentation. *Int. J Comput. Vision*, **70(2)**, 109-131, 2006.
8. Liu, L., at al.: Interactive separation of segmented bones in CT volumes using graph cut. *Proc. MICCAI*, 2008.
9. Freedman, D. and Zhang, T.: Interactive Graph Cut Based Segmentation With Shape Priors. *Proc. CVPR*, 2005.
10. Shim, H., at al.: Knee cartilage: efficient and reproducible segmentation on high-spatial-resolution MR images with the semiautomated graph-cut algorithm method. *Radiology*, **251(2)**, 548-556, 2009.
11. Li, K., at al.: Optimal surface segmentation in volumetric images-a graph-theoretic approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, **28(1)**, 119–134, 2006.
12. Yin, Y., et al.: LOGISMOS–layered optimal graph image segmentation of multiple objects and surfaces: Cartilage segmentation in the knee joint. *IEEE Trans. Med. Imag.*, **29(12)**, 2023-2037, 2010.
13. Kainmueller, D., et al.: Multi-Object Segmentation with Coupled Deformable Models. *Annals of BMVA*. 2009.
14. Zheng, Y., at al.: Four-Chamber Heart Modeling and Automatic Segmentation for 3-D Cardiac CT Volumes Using Marginal Space Learning and Steerable Features. *IEEE Trans. Med. Imag.*, **27**, 11, 1668–1681, 2008.
15. Tu, Z.: Probabilistic Boosting-Tree: Learning Discriminative Models for Classification, Recognition, and Clustering. *Proc. ICCV*, **2**, 1589 – 1596, 2005.
16. Zhang, J., at al.: Joint Real-Time Object Detection and Pose Estimation Using Probabilistic Boosting Network. *Proc. CVPR*, 2007.
17. Felzenszwalb, P. and Huttenlocher, D: Distance Transforms of Sampled Functions *Cornell Computing and Information Science*, 2004.
18. Delong, A. and Boykov, Y.: Globally Optimal Segmentation of Multi-Region Objects. *Proc. ICCV*, 2009.
19. Kolmogorov, V. and Zabin, R.: What Energy Functions Can Be Minimized Via Graph Cuts? *IEEE Trans. Pattern Anal. Mach. Intell.*, **26**, 2, 147-159, 2004.