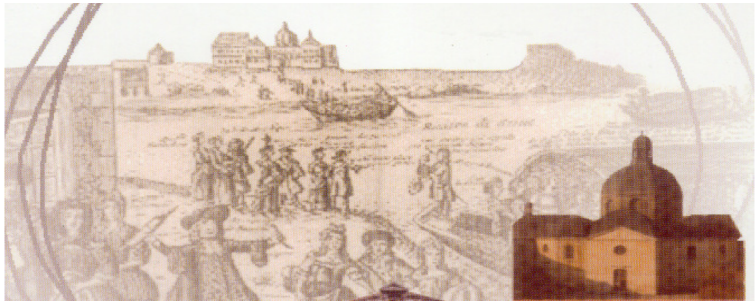
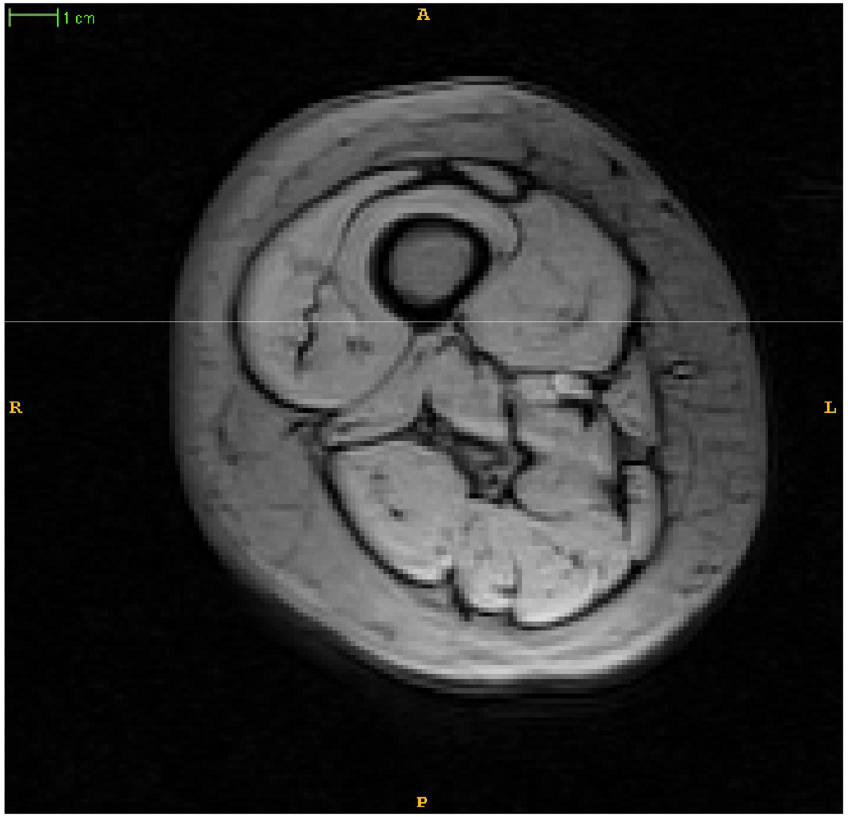
Automatized & Interactive Human Skeletal Muscle Segmentation

Noura Azzabou
30 April 2013

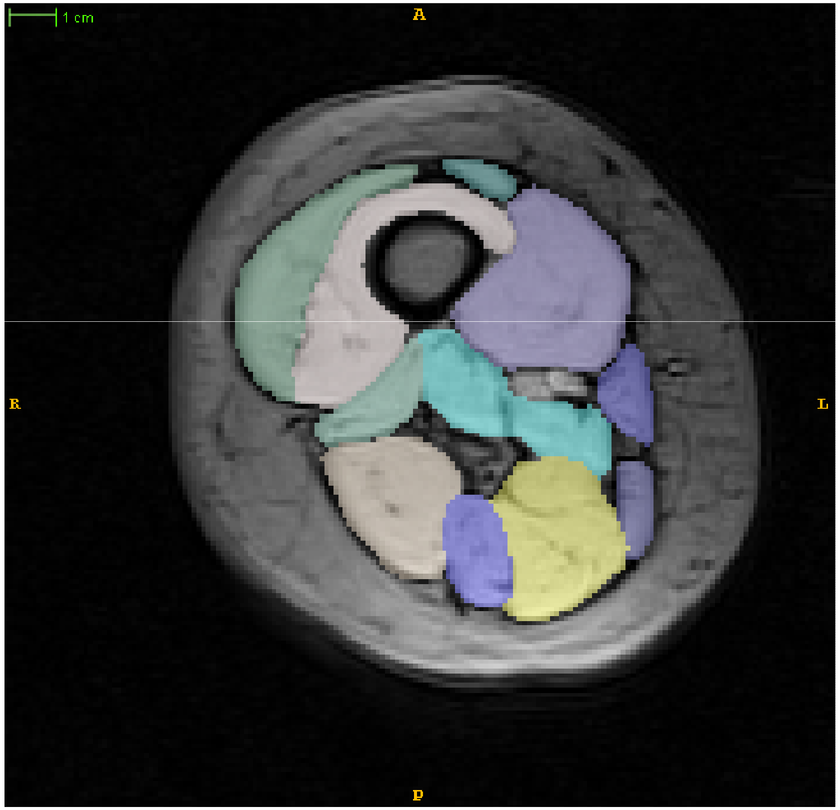


What is muscle segmentation ?

Axial slice of the thigh of a healthy volunteer (Dixon image)



A non segmented view



A manually segmented view

Purposes of muscle segmentation:

- To study each muscle selectively, for a better characterization of neuromuscular disorder.
- To access volumetric muscle changes.

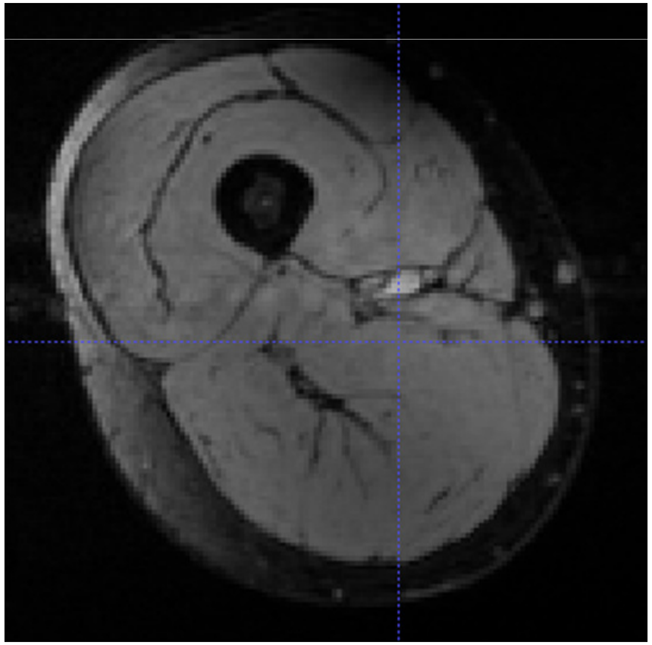
From manual to automated segmentation…

Today : segmentation of all muscles of a human thigh is :

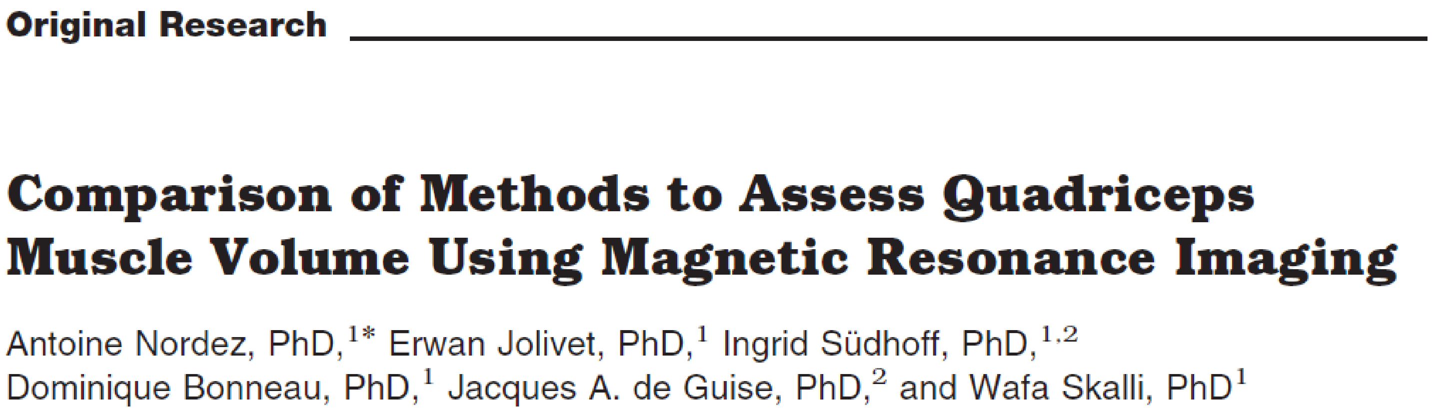
- Manual
- Extremely long (4 hours)
- Tedious
- Variable (Inter-operator volume variability: 3 %)

Why is automatic Segmentation challenging?

- Multi-object, partial contours, no distinctive texture
- Large inter-subject variations



Example of missing contours



Original Research

Comparison of Methods to Assess Quadriceps Muscle Volume Using Magnetic Resonance Imaging

Antoine Nordez, PhD,[1*] Erwan Jolivet, PhD,[1] Ingrid Südhoff, PhD,[1,2]
Dominique Bonneau, PhD,[1] Jacques A. de Guise, PhD,[2] and Wafa Skalli, PhD[1]

For comparison and volume assessement :

The vastus lateralis (VL), vastus intermedius (VI), vastus medialis (VM), and rectus femoris (RF) muscles of the 10 subjects were manually outlined for all of the slices by the same investigator Because substantial fusion may be found between VL and VI on some slices (3), these two muscles were outlined Together.

- The truncated cone formula [1] (30), which assumes a conical shape of the muscle between available slices:

$$MV = \sum_{n-1} \frac{e_i}{3} \times (CSA_i + CSA_{i+1} + \sqrt{CSA_i \times CSA_{i+1}}) \quad [1]$$

  Where n is the number of slices used, and e_i is the distance between available slices i and i+1
- The Cavalieri formula [2] (18), which assumes an ideal cylinder shape of the muscle between the available slices

$$MV = \sum_{n} e_i \times CSA_i \quad [2]$$

- A cubic spline interpolation was used for the relationship between CSA calculated on available slices and distance from insertion to estimate missing CSAs. The MV was then calculated using the Cavalieri formula [2].
- The deformation of a parametric specific object (DPSO) method (29), which could be decomposed in four steps. (i) Each muscle contour obtained on available slices was modeled as an equivalent ellipse. The centroid coordinates, local inertial coordinate systems, width and length were calculated for each available ellipse. (ii) Changes in these parameters along the muscle's principal axis were modeled using the cubic spline interpolation to estimate missing ellipses. Using all available and estimated ellipses, the resulting three-dimensional parametric object was reconstructed. (iii) The subject-specific volumetric muscle reconstruction (Fig. 1C) was determined by deforming the parametric object using a nonisotropic algorithm (31) and available muscle contours. (iv) The MV calculation was performed on this three-dimensional volumetric muscle reconstruction.

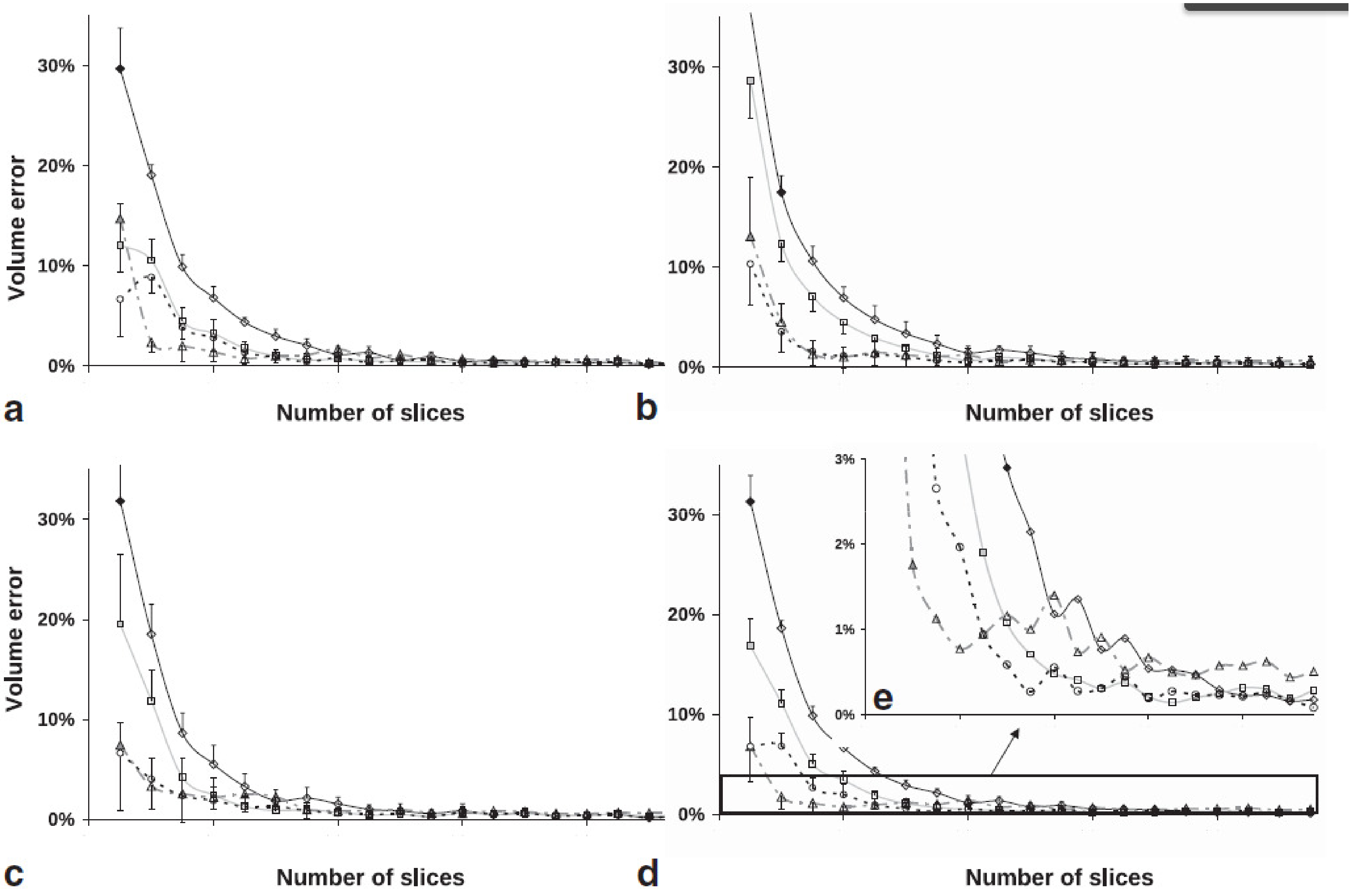


Figure 3. **a–d:** Error of *vastus lateralis* + *vastus intermedius* (a), *vastus medialis* (b), *rectus femoris* (c), and quadriceps (d) volume estimations using the truncated cones (diamonds), Cavalieri (squares), interpolation (triangles), deformation of parametric specific objects (circles) methods as functions of the number of available slices in respect to the MRI reference standard measure. The significant main interaction ($P < 0.001$) of the 4×19 (method × number of slices) ANOVA is described. **e:** A detail of (d).

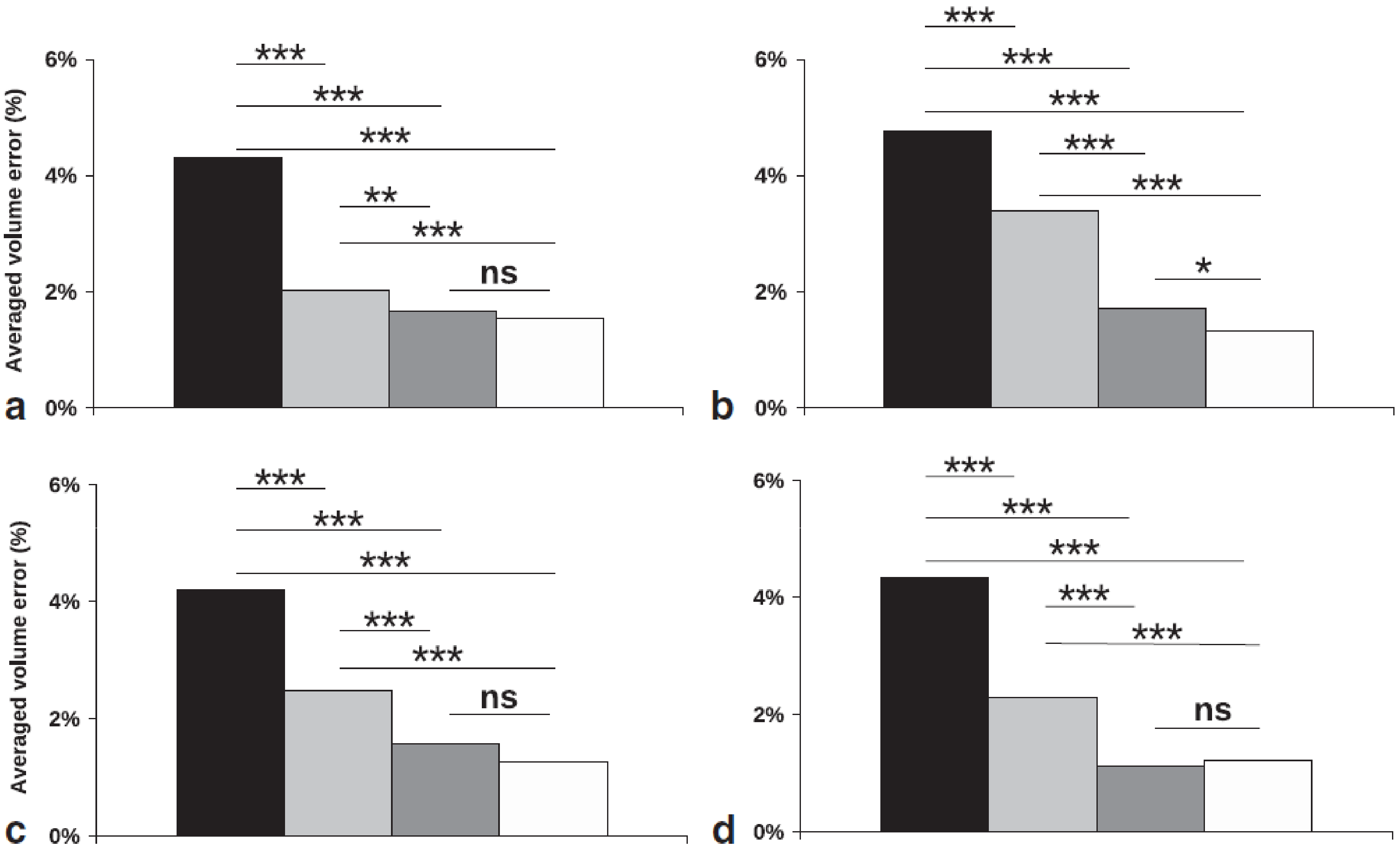
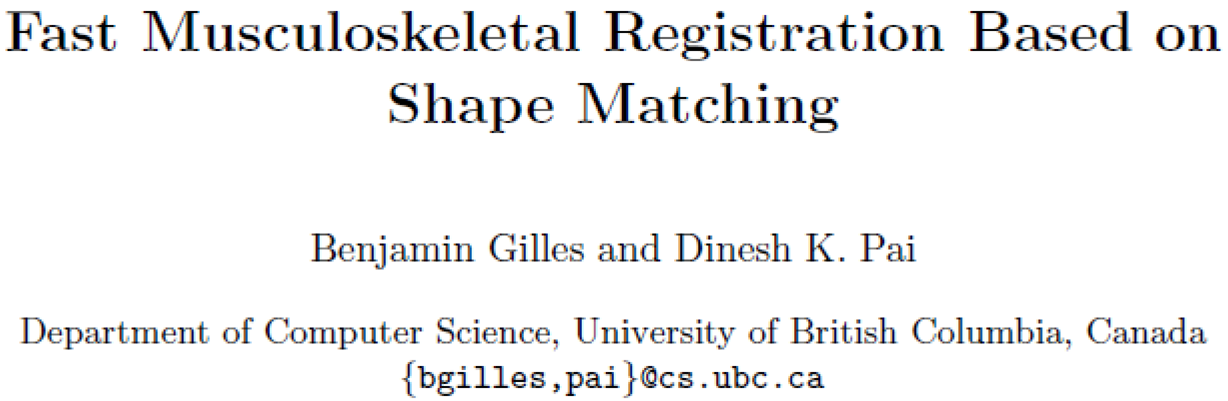


Figure 2. **a–d:** Average error of *vastus lateralis* + *vastus intermedius* (a), *vastus medialis* (b), *rectus femoris* (c), and quadriceps (d) muscle volume estimations using the truncated cones (black shading), Cavalieri (light shading), interpolation (medium shading), deformation of parametric specific objects (white) methods in respect to the MRI reference standard measure. The significant main effect ($P < 0.001$) of method in the 4 × 19 (method × number of slices) ANOVA is presented. *$P < 0.05$; **$P < 0.01$; ***$P < 0.001$; ns: nonsignificant

- The techniques described in this work are dedicated to muscle volume assessement and does not offer an accurate segmentation technique.

- Accurate volume estimation requires the manual segmentation of an important number of slices

Need of accurate segmentation with little intreaction



Fast Musculoskeletal Registration Based on Shape Matching

Benjamin Gilles and Dinesh K. Pai

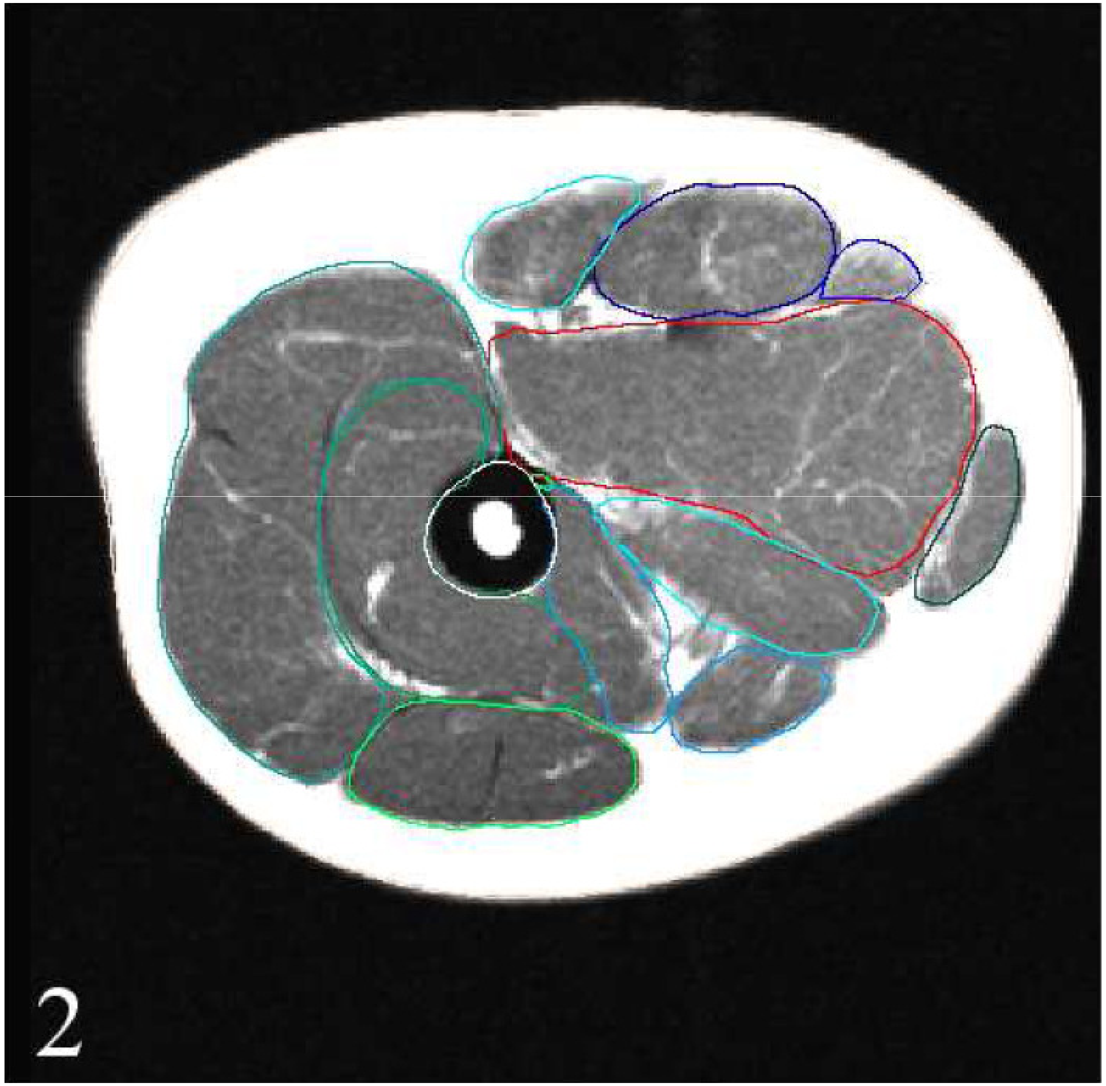
Department of Computer Science, University of British Columbia, Canada
{bgilles,pai}@cs.ubc.ca

Automatic muscle group segmentation through the registration of a model on the image and using as internal forces an average of rigid local transforms.

Gilles et al (MICCAI 2008)

Model construction

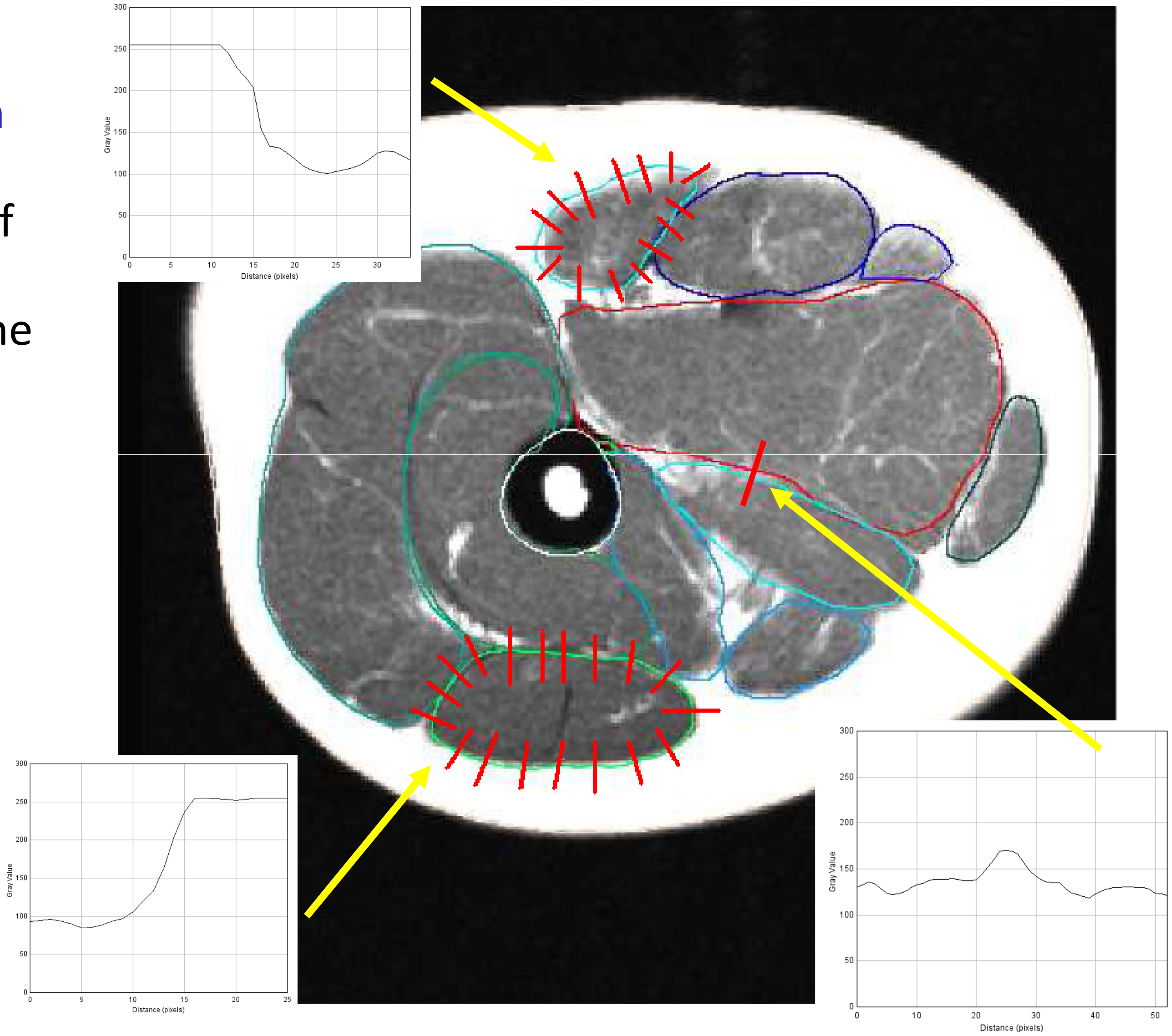
1- manually segment a given reference image.



Gilles et al (MICCAI 2008)

Model construction

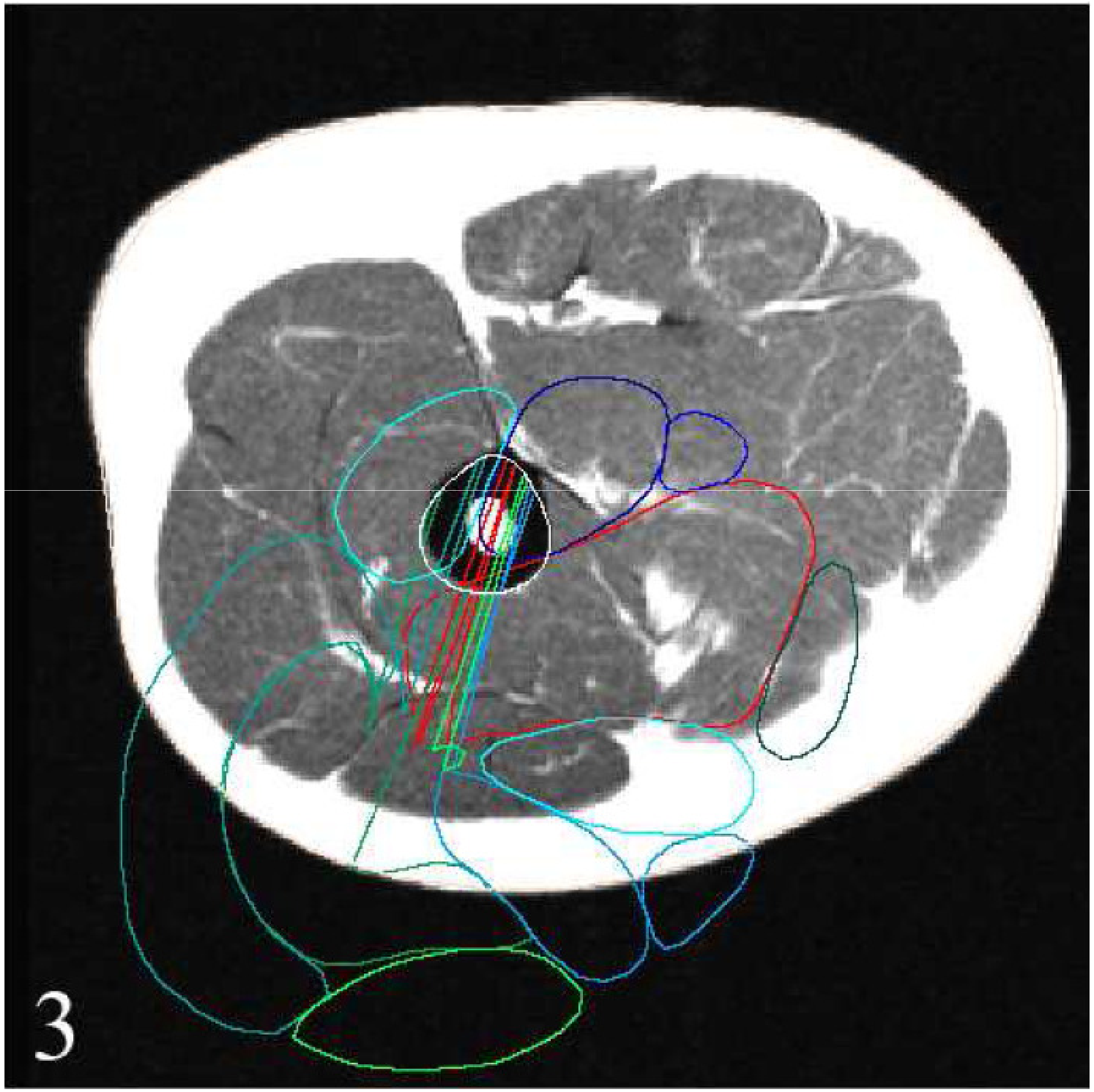
2- For each vertex of the mesh, Learn intensity profile in the normal direction



Gilles et al (MICCAI 2008)

Shape registration :

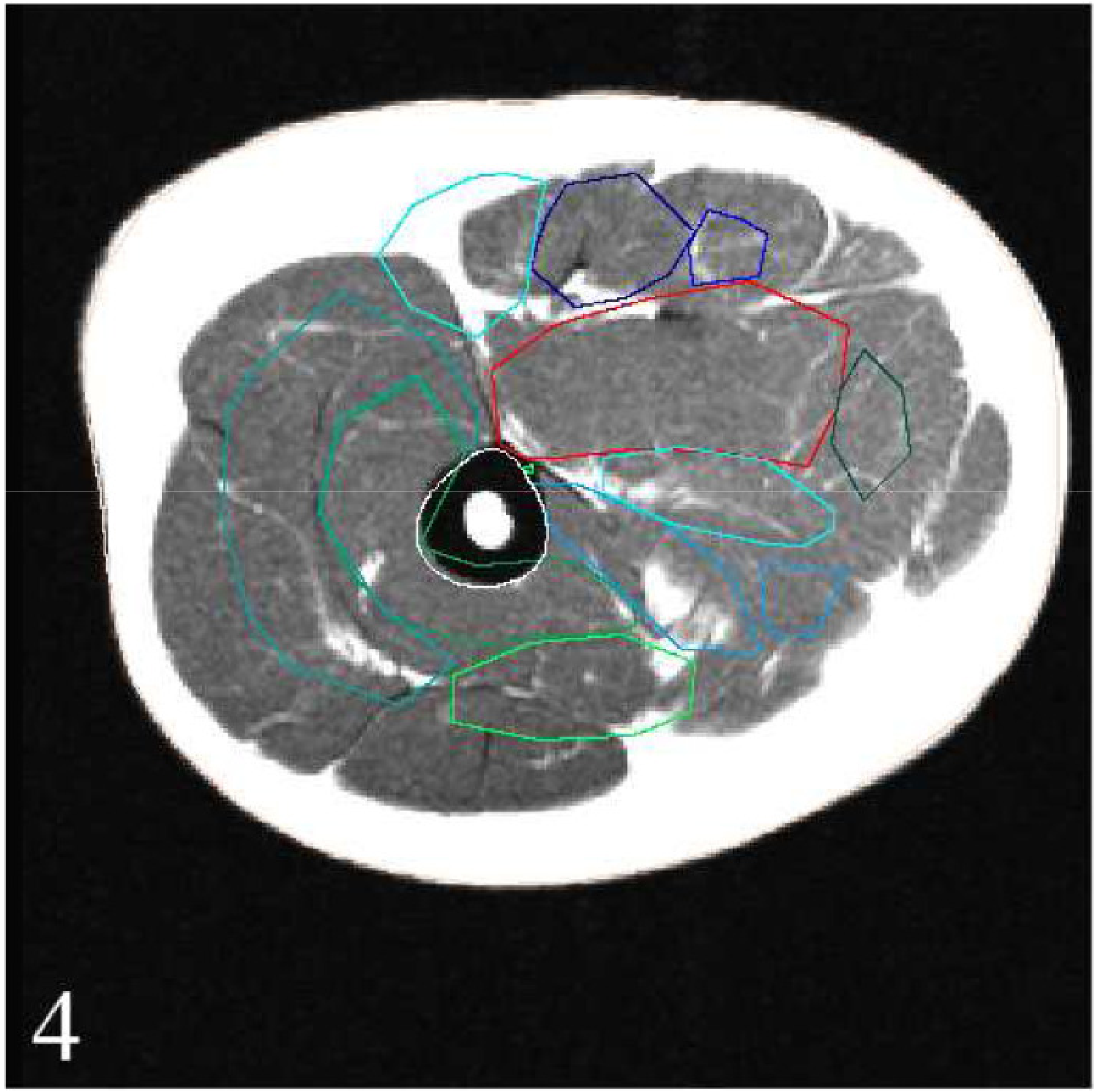
1- First rough registration based on bone position



Gilles et al (MICCAI 2008)

Shape registration :

1- First rough registration based on bone position



Gilles et al (MICCAI 2008)

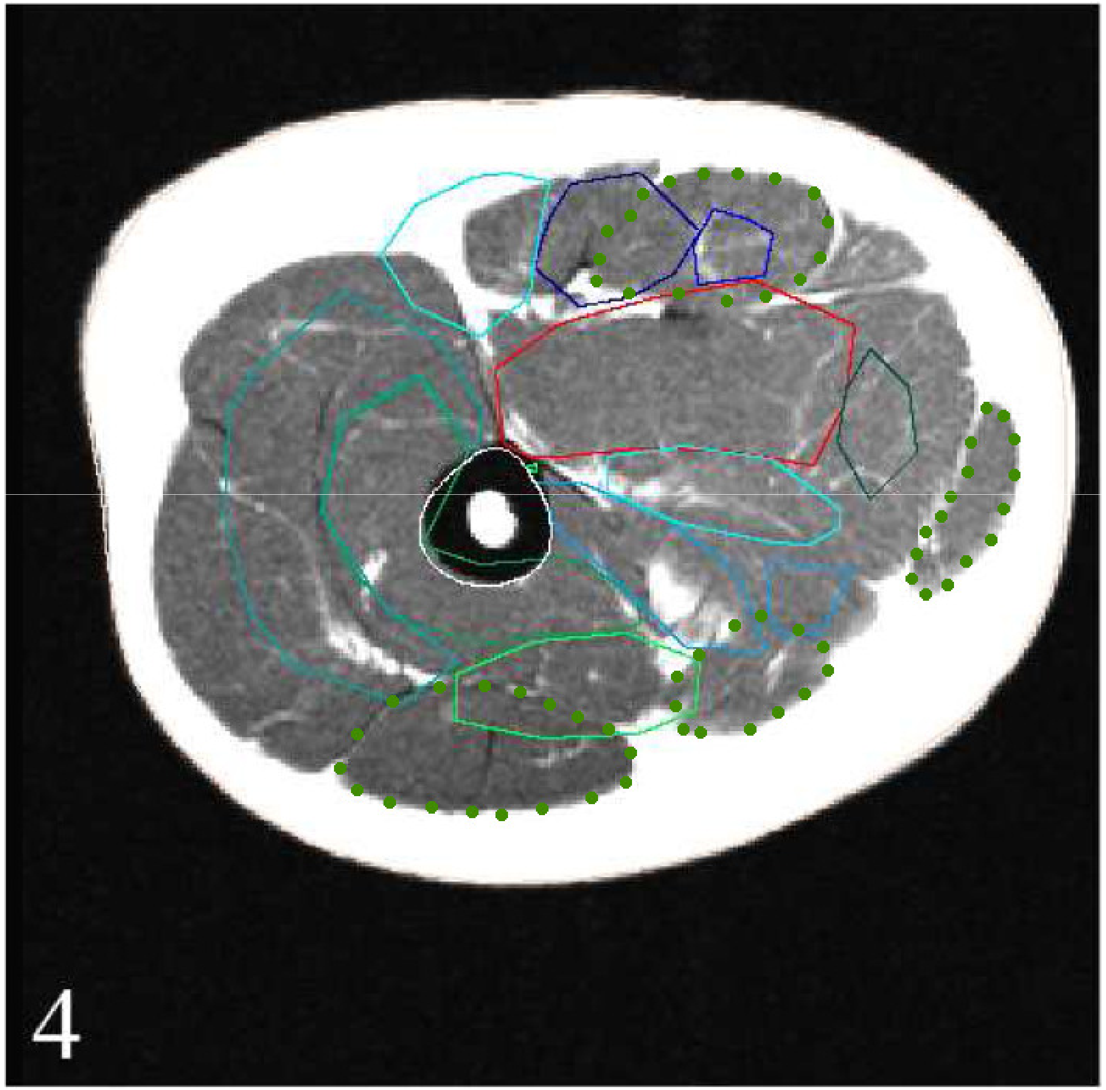
Shape registration :

2- For each vertex of the mesh search for :

- The voxel that have the closest intensity profile to the reference one.

- The voxels that have a maximum gradient

The target point is the mean of these two point.

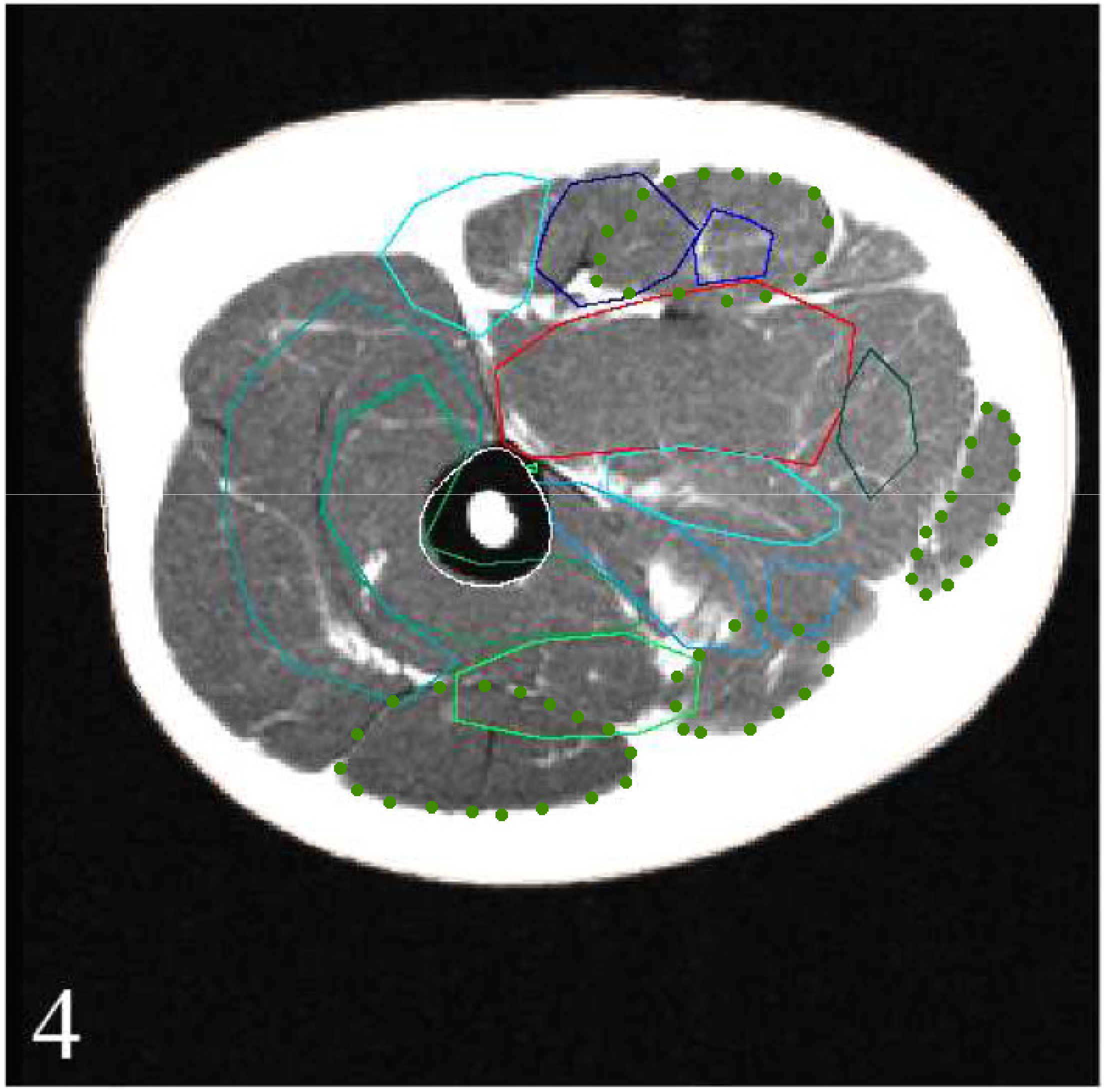


Gilles et al (MICCAI 2008)

Shape registration :

Define external forces as the forces that will deform the mesh toward the target points defined previously

!!! Important deformation may lead to a non smooth contours and to a solution that not correspond to the expected form of a muscle



Gilles et al (MICCAI 2008)

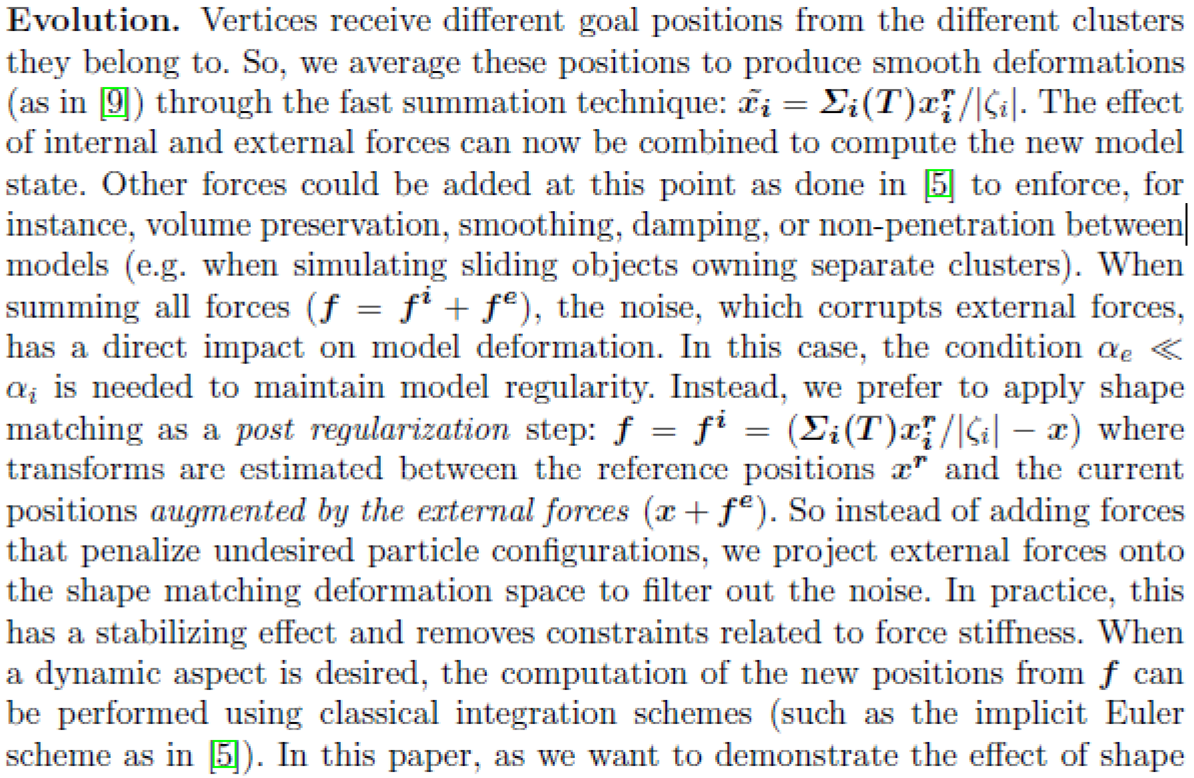
Shape registration : internal forces

Aims to constrain the deformations of the shape due to external forces and to keep it similar to the reference one.

Performs a mapping of the model vertex to the target pixels based on local rigid transforms.

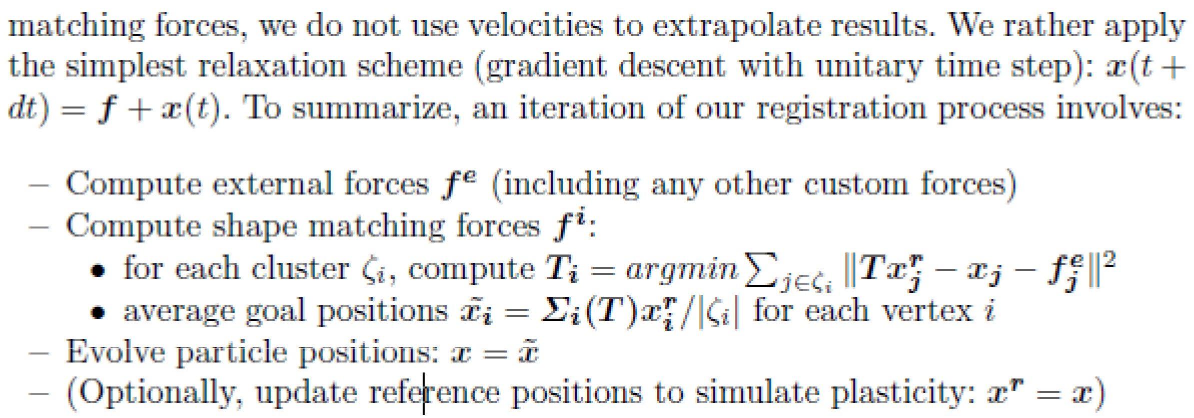
➔ To insure smoothness of deformation, the neighboring vertexes in the reference model are clustered and each cluster is deformed using a rigid transform to much the target points.

Gilles et al (MICCAI 2010)



Evolution. Vertices receive different goal positions from the different clusters they belong to. So, we average these positions to produce smooth deformations (as in [9]) through the fast summation technique: $\tilde{\boldsymbol{x}}_{\boldsymbol{i}} = \boldsymbol{\Sigma_i}(\boldsymbol{T})\boldsymbol{x}^{\boldsymbol{r}}_{\boldsymbol{i}}/|\zeta_i|$. The effect of internal and external forces can now be combined to compute the new model state. Other forces could be added at this point as done in [5] to enforce, for instance, volume preservation, smoothing, damping, or non-penetration between models (e.g. when simulating sliding objects owning separate clusters). When summing all forces ($\boldsymbol{f} = \boldsymbol{f}^{\boldsymbol{i}} + \boldsymbol{f}^{\boldsymbol{e}}$), the noise, which corrupts external forces, has a direct impact on model deformation. In this case, the condition $\alpha_e \ll \alpha_i$ is needed to maintain model regularity. Instead, we prefer to apply shape matching as a *post regularization* step: $\boldsymbol{f} = \boldsymbol{f}^{\boldsymbol{i}} = (\boldsymbol{\Sigma_i}(\boldsymbol{T})\boldsymbol{x}^{\boldsymbol{r}}_{\boldsymbol{i}}/|\zeta_i| - \boldsymbol{x})$ where transforms are estimated between the reference positions $\boldsymbol{x}^{\boldsymbol{r}}$ and the current positions *augmented by the external forces* ($\boldsymbol{x} + \boldsymbol{f}^{\boldsymbol{e}}$). So instead of adding forces that penalize undesired particle configurations, we project external forces onto the shape matching deformation space to filter out the noise. In practice, this has a stabilizing effect and removes constraints related to force stiffness. When a dynamic aspect is desired, the computation of the new positions from $\boldsymbol{f}$ can be performed using classical integration schemes (such as the implicit Euler scheme as in [5]). In this paper, as we want to demonstrate the effect of shape

Gilles et al (MICCAI 2008)



matching forces, we do not use velocities to extrapolate results. We rather apply the simplest relaxation scheme (gradient descent with unitary time step): $\boldsymbol{x}(t+dt) = \boldsymbol{f} + \boldsymbol{x}(t)$. To summarize, an iteration of our registration process involves:

- Compute external forces $\boldsymbol{f^e}$ (including any other custom forces)
- Compute shape matching forces $\boldsymbol{f^i}$:
  - for each cluster ζ_i, compute $\boldsymbol{T_i} = argmin \sum_{j\in\zeta_i} \|\boldsymbol{T x_j^r} - \boldsymbol{x_j} - \boldsymbol{f_j^e}\|^2$
  - average goal positions $\tilde{\boldsymbol{x}}_{\boldsymbol{i}} = \boldsymbol{\Sigma_i}(\boldsymbol{T})\boldsymbol{x_i^r}/|\zeta_i|$ for each vertex i
- Evolve particle positions: $\boldsymbol{x} = \tilde{\boldsymbol{x}}$
- (Optionally, update reference positions to simulate plasticity: $\boldsymbol{x^r} = \boldsymbol{x}$)

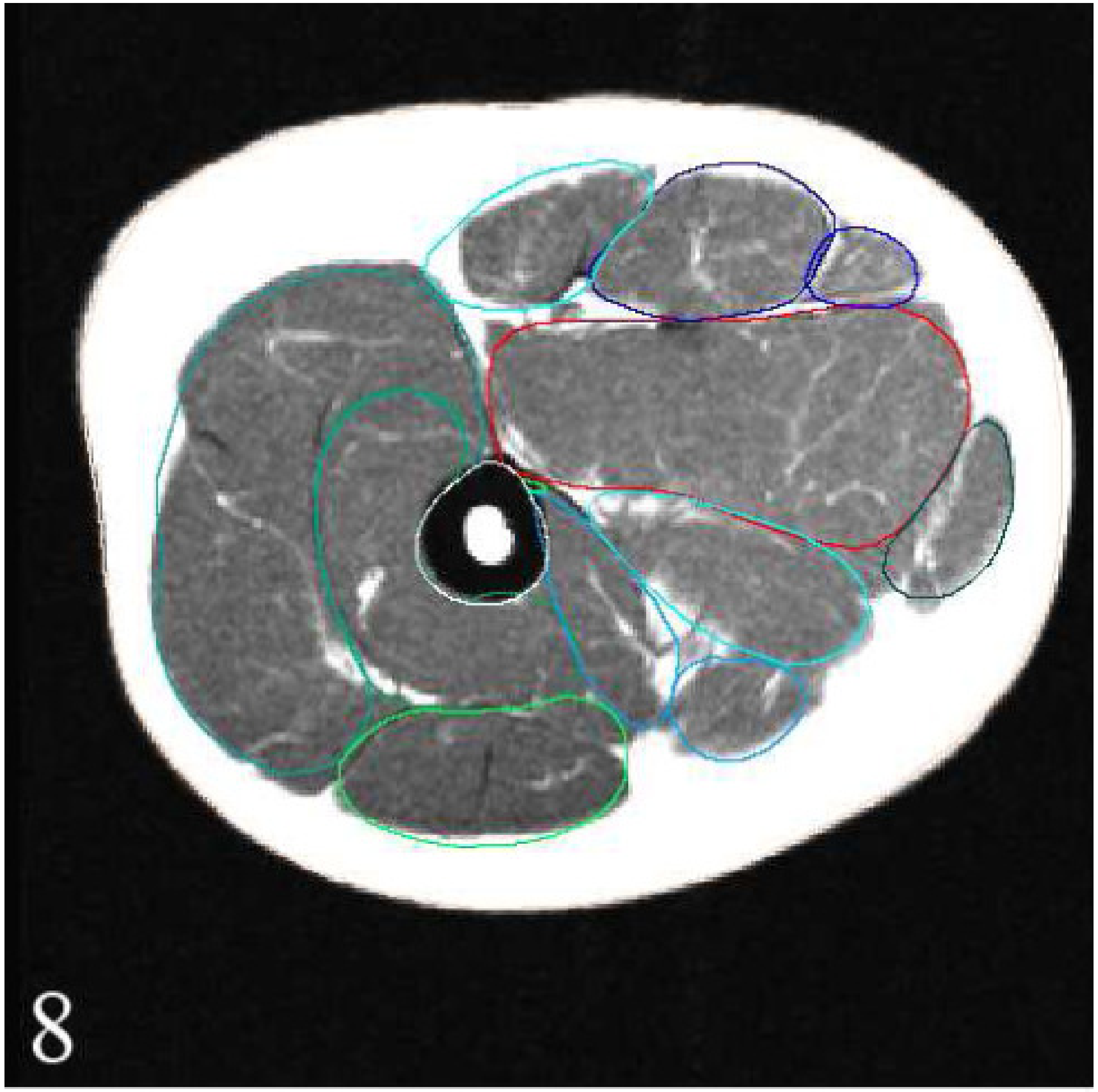
Gilles et al (MICCAI 2008)

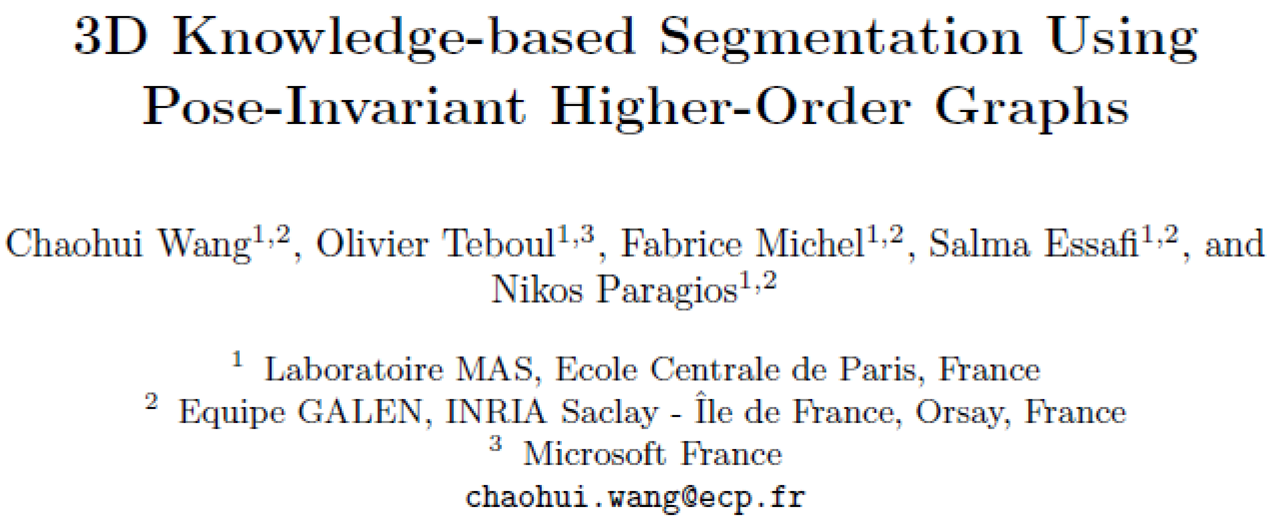
Segmentation result

++ among the first solutions for muscle segmentation

++ many muscles are well delineated

-- inspite of a good contrast some segmentation error are observed and they are due to the model





3D Knowledge-based Segmentation Using Pose-Invariant Higher-Order Graphs

Chaohui Wang[1,2], Olivier Teboul[1,3], Fabrice Michel[1,2], Salma Essafi[1,2], and
Nikos Paragios[1,2]

[1] Laboratoire MAS, Ecole Centrale de Paris, France
[2] Equipe GALEN, INRIA Saclay - Île de France, Orsay, France
[3] Microsoft France
`chaohui.wang@ecp.fr`

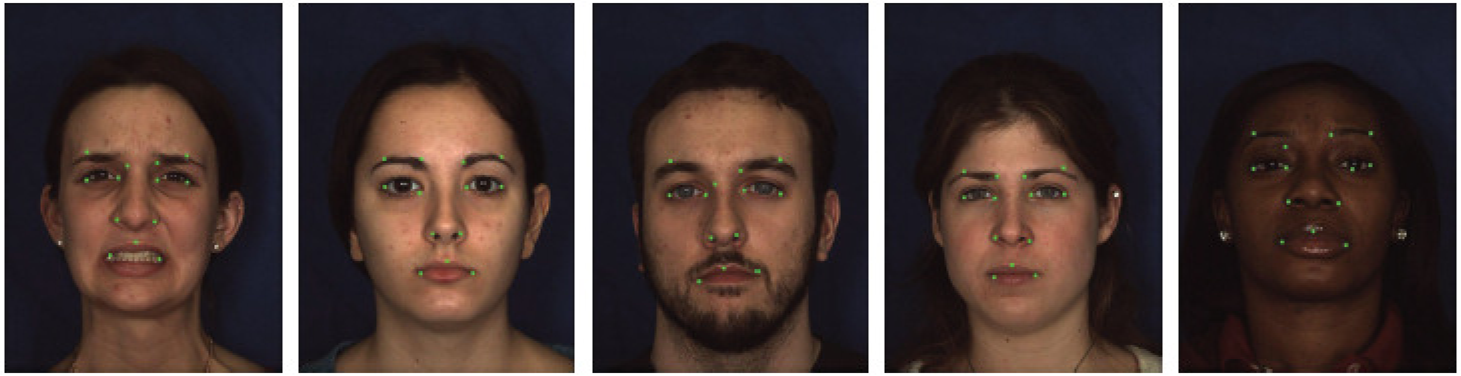
1- Shape Model is discrete and based on landmarks.

2- Segmentation is equivalent to a registration of set of landMark on the image to segment.

3- Only one muscle is segmented.

Wang et al (MICCAI 2010)

Model learning : shape



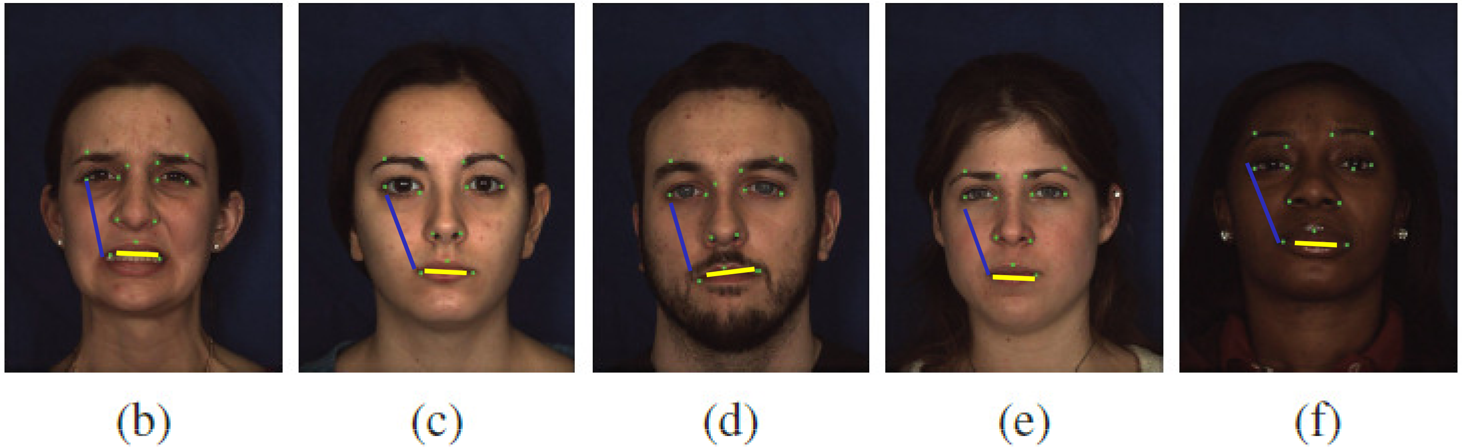
(b) (c) (d) (e) (f)

1- Register all the training image together

2- Find a consistent set of landmarks, between all the training image

Wang et al (MICCAI 2010)

Model learning : shape



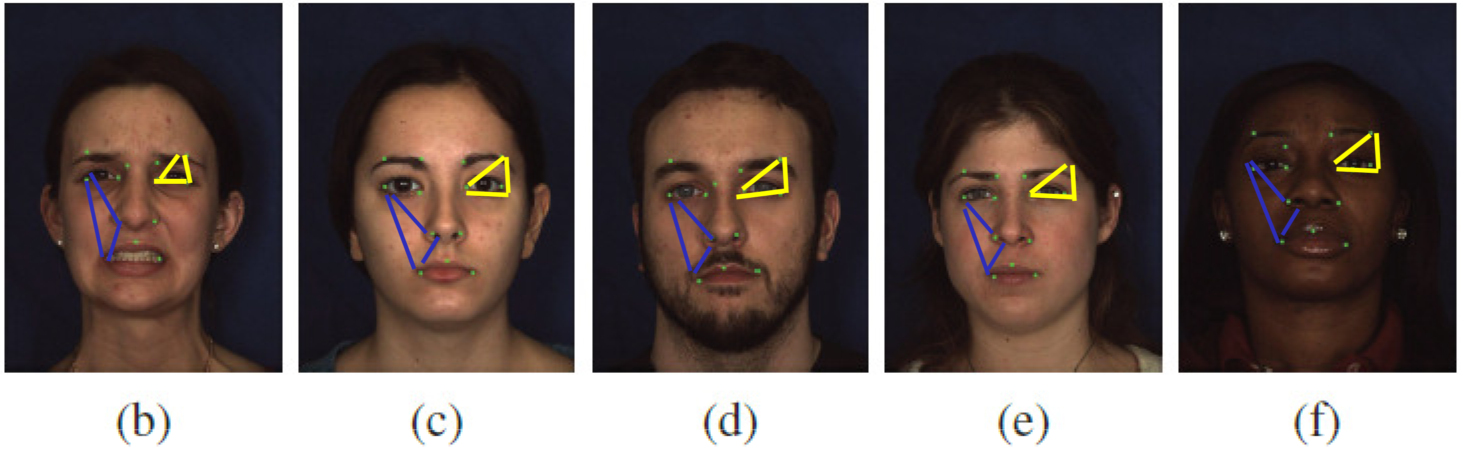
1- Register all the training image together
2- Find a consistent set of landmarks, between all the training image

How to describe the model :

1- Consider the length of couple of landMarks (pb de scale invariance)

Wang et al (MICCAI 2010)

Model learning : shape


(b) (c) (d) (e) (f)

1- Register all the training image together
2- Find a consistent set of landmarks, between all the training image

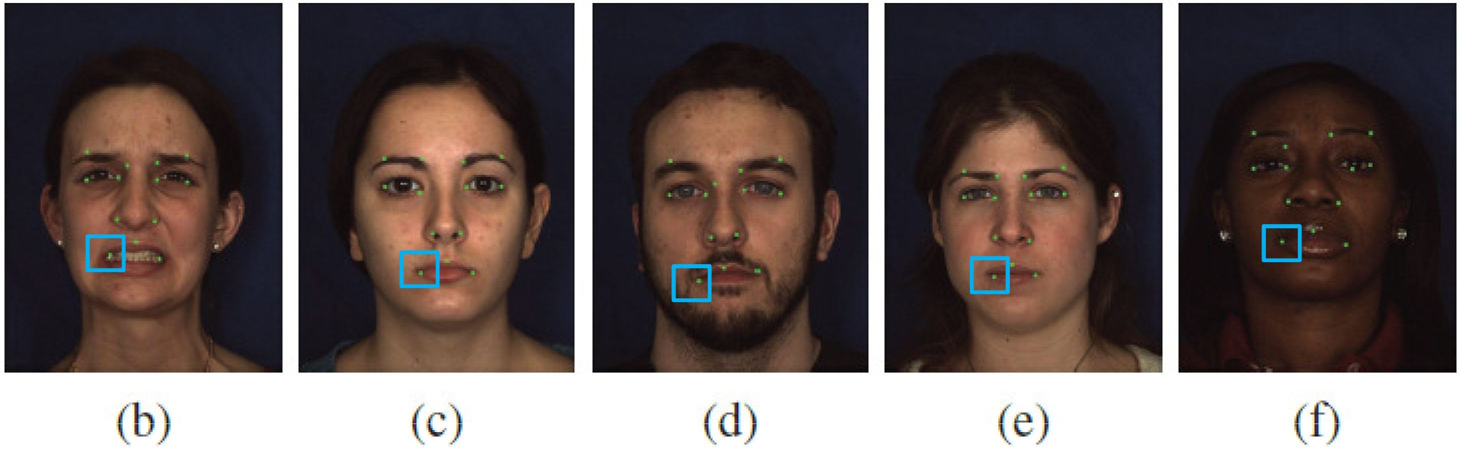
How to describe the model :

1- Consider the length of couple of landMarks (pb de scale invariance)
2- Consider the normalized length of all triplets of landMark.

Over the training set we can probability distribution function of these distances.

Wang et al (MICCAI 2010)

Model learning : appearance


(b) (c) (d) (e) (f)

1- In addition to geometrical properties, local appearance properties of each landmark are learnt.

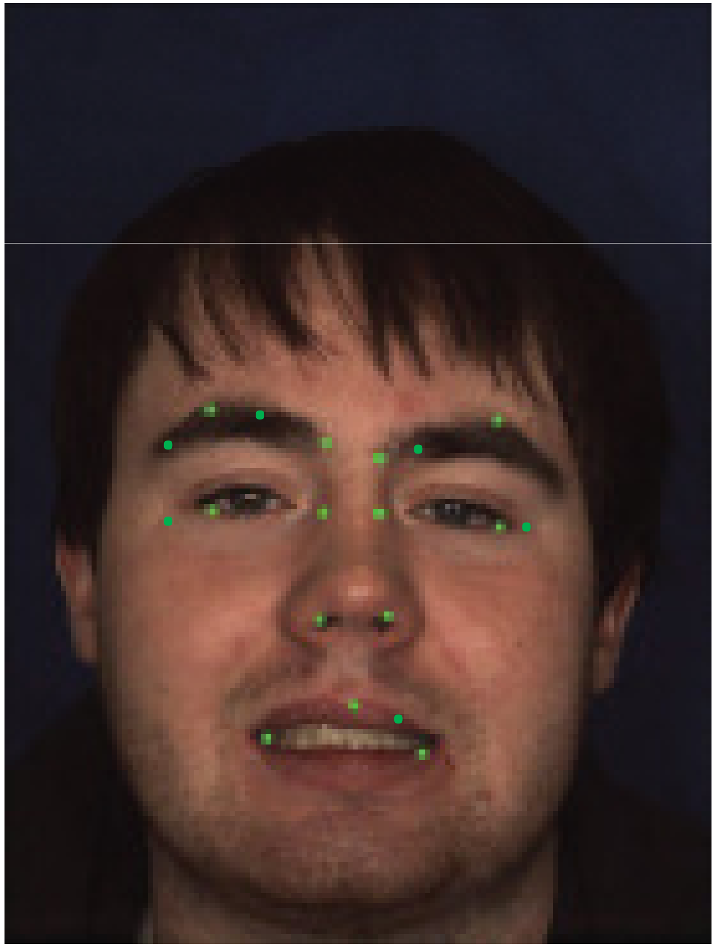
Wang et al (MICCAI 2010)

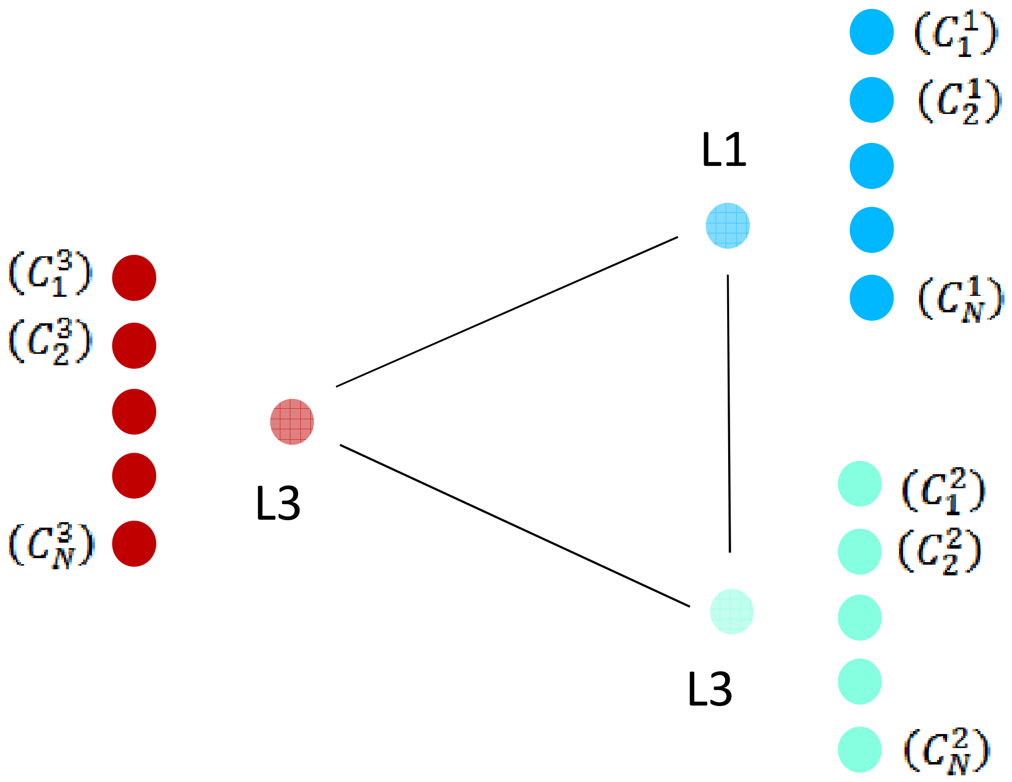
Segmentation ➔ a correspondence problem that requires detection for the model points in the image or finding a set of correspondences for each landmarks.

1- For a new image to segment and for each landmark, find the corresponding one using block matching technique.

➔If the best much is selected, we can find many false positive and the shape model is not respected.

➔ consider many candidates that have similar appearance and then select the ones that they are close to the shape model





Assume we have M landmarks, the matching operation amounts to finding the set of label $(C_i^L)_{i<N,L<M}$ that minimize

$$E = \sum_{k=1}^{M} diff(P_{k,}^{L} P_i^{C}) + \lambda \sum_{all\ triplets} p_{shape\ model}(C_i^{l1}, C_j^{l2}, C_k^{l3})$$

Use discrete optimization technique applied to high order Markov field [1].

[1] N. Komodakis, G. Tziritas, and N. Paragios. Performance vs computational efficiency for optimizing single and dynamic mrfs: Setting the state of the art with primal dual strategies. Comput. Vis. Image Und. (CVIU), 112(1):14–29, 2008

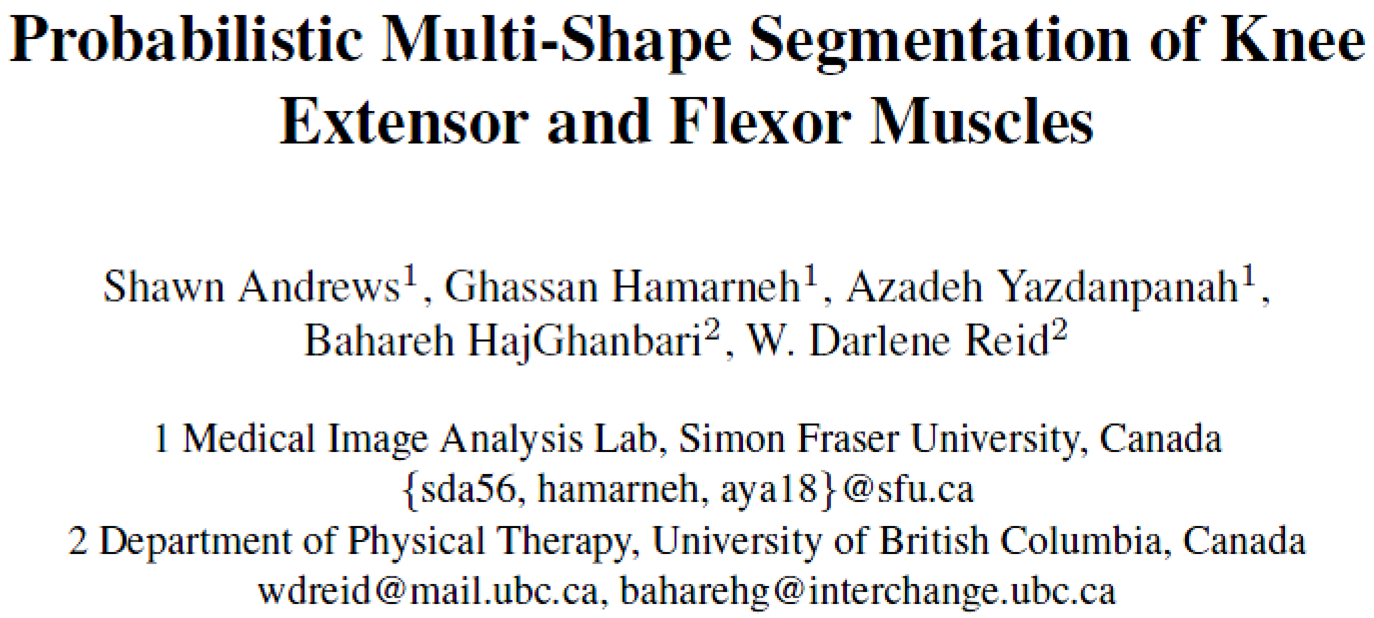
Wang et al (MICCAI 2010)

Conclusions :

- No segmentation results are provided, the evaluation was expressed in terms of how resulting landmarks are close to the manually placed landmark.

- Only one muscle is segmented but this technqiue could be easily extended to deal with several muscle segmentations.

- It is hard to imagine that a software can find in a consistent manner landmarks in different volumes, because muscle surface is very homogeneous and smooth.

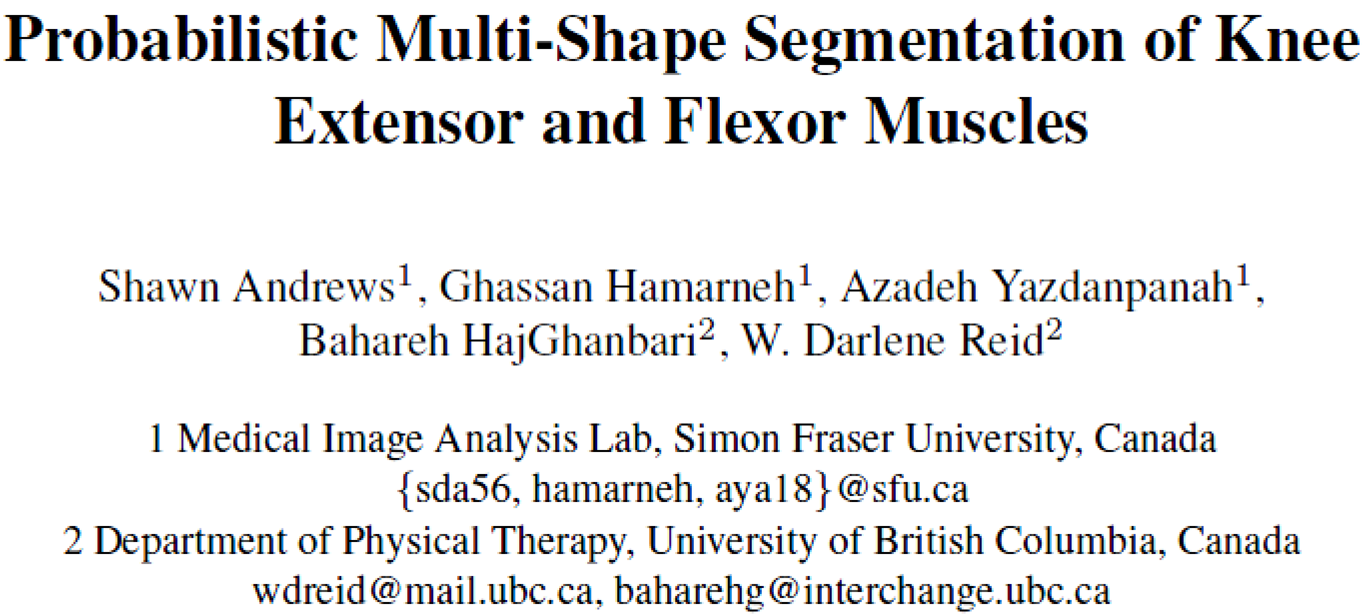


Probabilistic Multi-Shape Segmentation of Knee Extensor and Flexor Muscles

Shawn Andrews[1], Ghassan Hamarneh[1], Azadeh Yazdanpanah[1],
Bahareh HajGhanbari[2], W. Darlene Reid[2]

1 Medical Image Analysis Lab, Simon Fraser University, Canada
{sda56, hamarneh, aya18}@sfu.ca
2 Department of Physical Therapy, University of British Columbia, Canada
wdreid@mail.ubc.ca, baharehg@interchange.ubc.ca

1- Construct a Model using PCA
2- Align the model to the image using two information
    2.1 – image contours obtained using an edge detector
    2.2 – intensity model to detect, background, subcuteneous fat and muscles.



Probabilistic Multi-Shape Segmentation of Knee Extensor and Flexor Muscles

Shawn Andrews[1], Ghassan Hamarneh[1], Azadeh Yazdanpanah[1],
Bahareh HajGhanbari[2], W. Darlene Reid[2]

1 Medical Image Analysis Lab, Simon Fraser University, Canada
{sda56, hamarneh, aya18}@sfu.ca
2 Department of Physical Therapy, University of British Columbia, Canada
wdreid@mail.ubc.ca, baharehg@interchange.ubc.ca

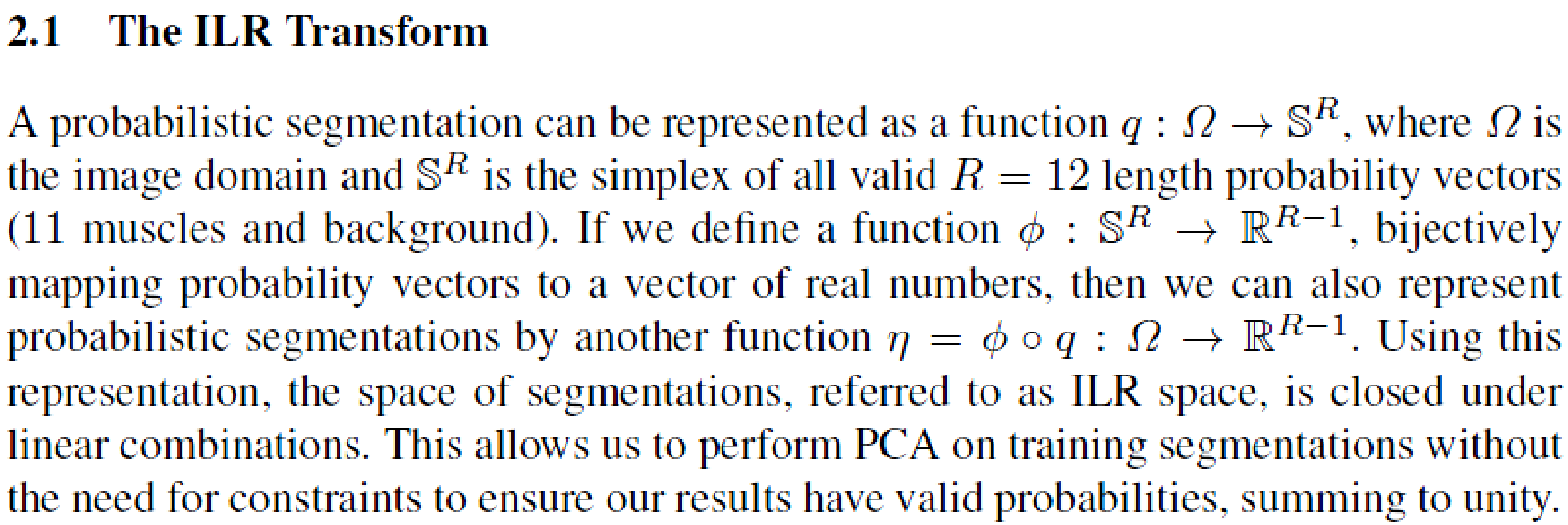
Model definition : A probabilistic map that assigns for each pixel the probability of belonging to one specific muscles.

Andrews et al (MICCAI 2011)

Model construction :

1- realign the different training segmentation using the anatomical images.

2- Apply the ILR transform which is a projection of the probability vector in a multi-dimensional with real value and lower dimension.



2.1 The ILR Transform

A probabilistic segmentation can be represented as a function $q : \Omega \rightarrow \mathbb{S}^R$, where Ω is the image domain and $\mathbb{S}^R$ is the simplex of all valid $R = 12$ length probability vectors (11 muscles and background). If we define a function $\phi : \mathbb{S}^R \rightarrow \mathbb{R}^{R-1}$, bijectively mapping probability vectors to a vector of real numbers, then we can also represent probabilistic segmentations by another function $\eta = \phi \circ q : \Omega \rightarrow \mathbb{R}^{R-1}$. Using this representation, the space of segmentations, referred to as ILR space, is closed under linear combinations. This allows us to perform PCA on training segmentations without the need for constraints to ensure our results have valid probabilities, summing to unity.

Andrews et al (MICCAI 2011)

Model construction :

1- realign the different training segmentation using the anatomical images.

2- Apply the ILR transform which is a projection of the probability vector in a multi-dimensional with real value and lower dimension.

3- performing a PCA on the training sample, the shape is parameterized as :

$$\eta(\gamma) = \eta_0 + \Gamma\gamma$$

4- Shape energy term enforces a Mahalanobis type penalty to allow components corresponding to eigenmodes of greater variance to vary more

$$E_{Shape}(\gamma) = \gamma^T \Lambda^{-1} \gamma \ .$$

Andrews et al (MICCAI 2011)

Registration of the model to the image

1- extract image contours

Slice Intensities

Thresholded Intensities

Curvature: c

Curvature: $\bar{c}$

Boundary: h

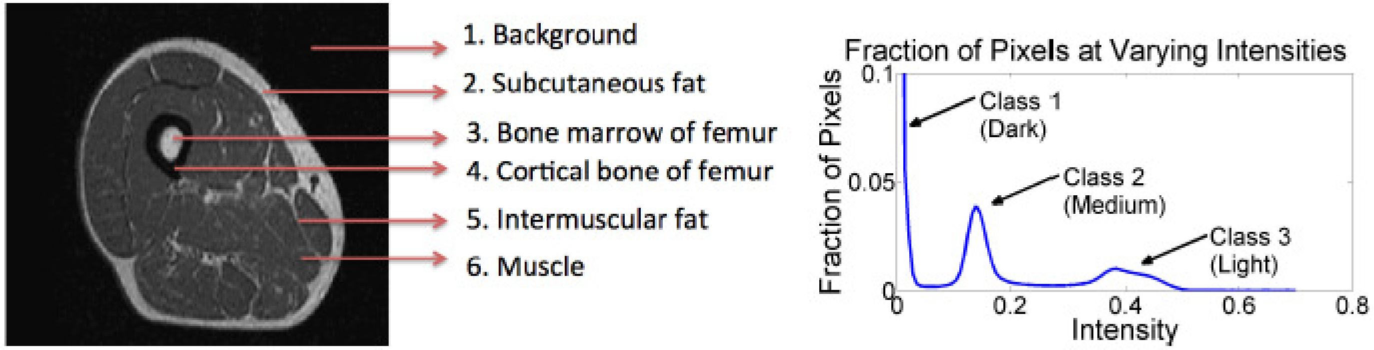
Fig. 3: Extracted image information. From left to right, we see for a thigh slice: the intensity values; the approximation of the muscle regions extracted by intensity thresholding; the curvature values c and $\bar{c}$; and the boundary estimate h.

$$E_{BDY}(\gamma) = \sum_{r=1}^{R-1} \sum_{x \in \Omega} (1 - h(x)) |\nabla_x \eta|^2 \; . \qquad (5)$$

E_{BDY} allows the segmentation to have large gradient only on voxels deemed likely to be part of a boundary, otherwise the gradient is penalized. Note once again that E_{BDY} is convex in η and thus in γ. Also note that $\nabla_x \eta$ measures the rate of change between the probabilities of neighboring voxels measured via the Aitchison distance.

Andrews et al (MICCAI 2011)

Registration of the model to the image



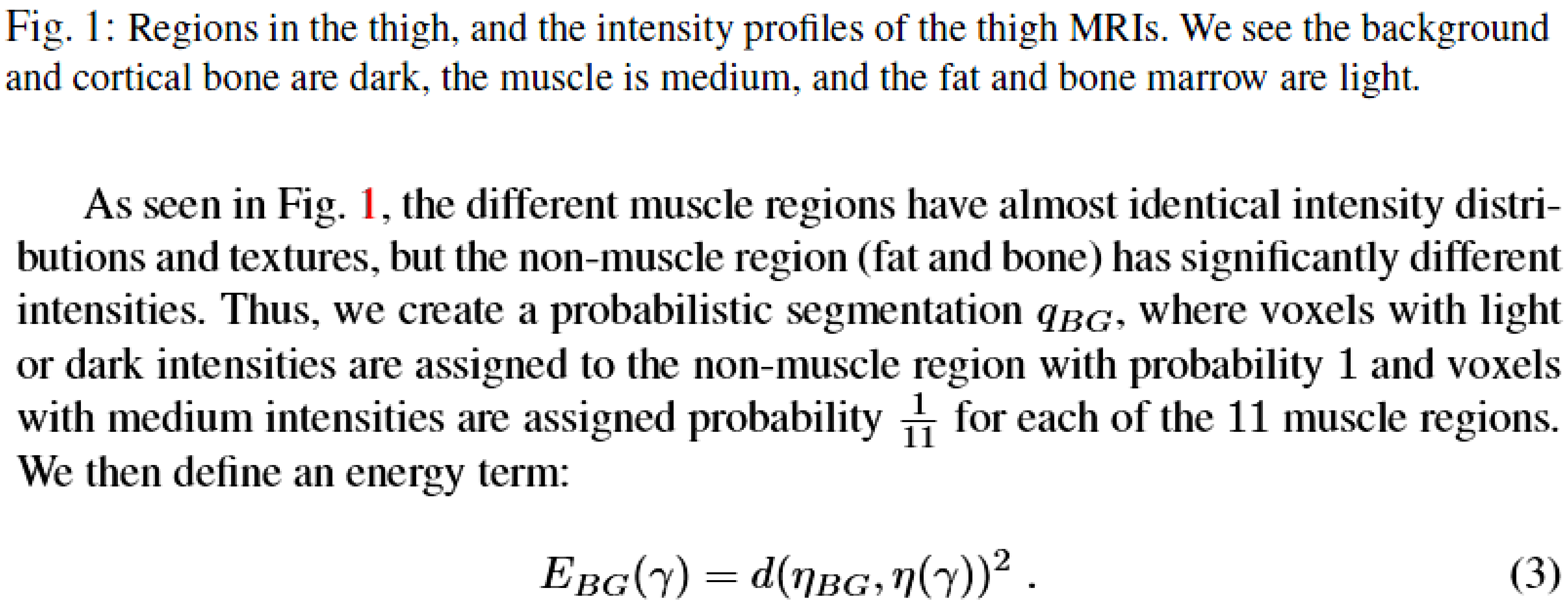


Fig. 1: Regions in the thigh, and the intensity profiles of the thigh MRIs. We see the background and cortical bone are dark, the muscle is medium, and the fat and bone marrow are light.

As seen in Fig. 1, the different muscle regions have almost identical intensity distributions and textures, but the non-muscle region (fat and bone) has significantly different intensities. Thus, we create a probabilistic segmentation q_{BG}, where voxels with light or dark intensities are assigned to the non-muscle region with probability 1 and voxels with medium intensities are assigned probability $\frac{1}{11}$ for each of the 11 muscle regions. We then define an energy term:

$$E_{BG}(\gamma) = d(\eta_{BG}, \eta(\gamma))^2 \ . \tag{3}$$

Andrews et al (MICCAI 2011)

Registration of the model to the image

Our energy is a linear combination of the convex energy terms (3), (5), and (6):

$$E(\gamma) = \lambda_1 E_{BG}(\gamma) + \lambda_2 E_{BDY}(\gamma) + \lambda_3 E_{Shape}(\gamma) \ . \quad (7)$$

Once our energy is constructed, we segment a novel image by finding $\gamma_{min} = \operatorname{argmin}_\gamma E(\gamma)$, and taking $q_{seg} = \phi^{-1}(\eta_0 + \Gamma\gamma_{min})$ as our final probabilistic segmentation.

Results & Validation

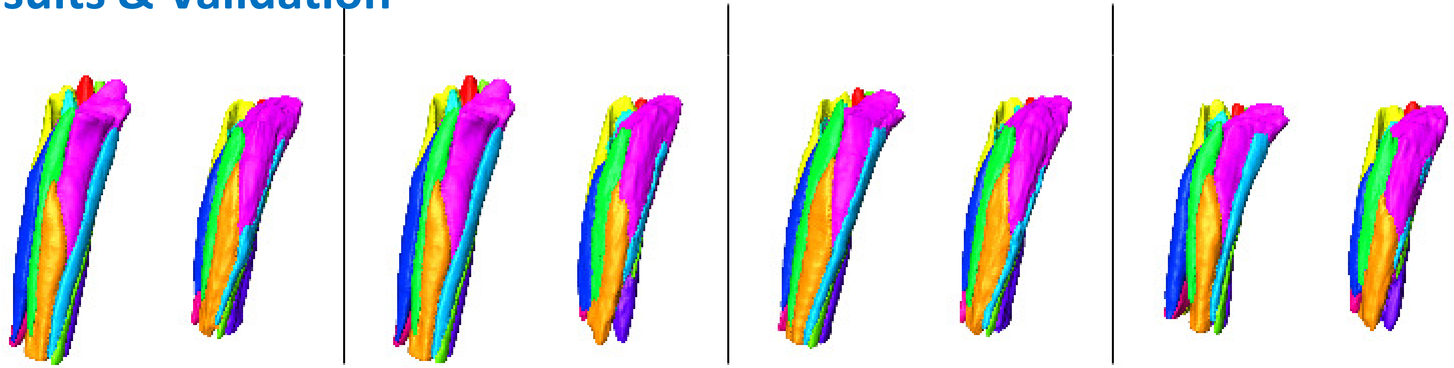


Fig. 5: (Color Figure) A comparison between several of the GT segmentations (left side of each pair) and the segmentations generated by our method (right side of each pair). The left two pairs are from COPD patients and the right two pairs are from non-COPD patients. DSC scores for each pair are 0.94, 0.91, 0.95, and 0.93.

Andrews et al (MICCAI 2011)

Results & Validation

Probabilistic Thigh Muscle Segmentation Using Shape Priors 7

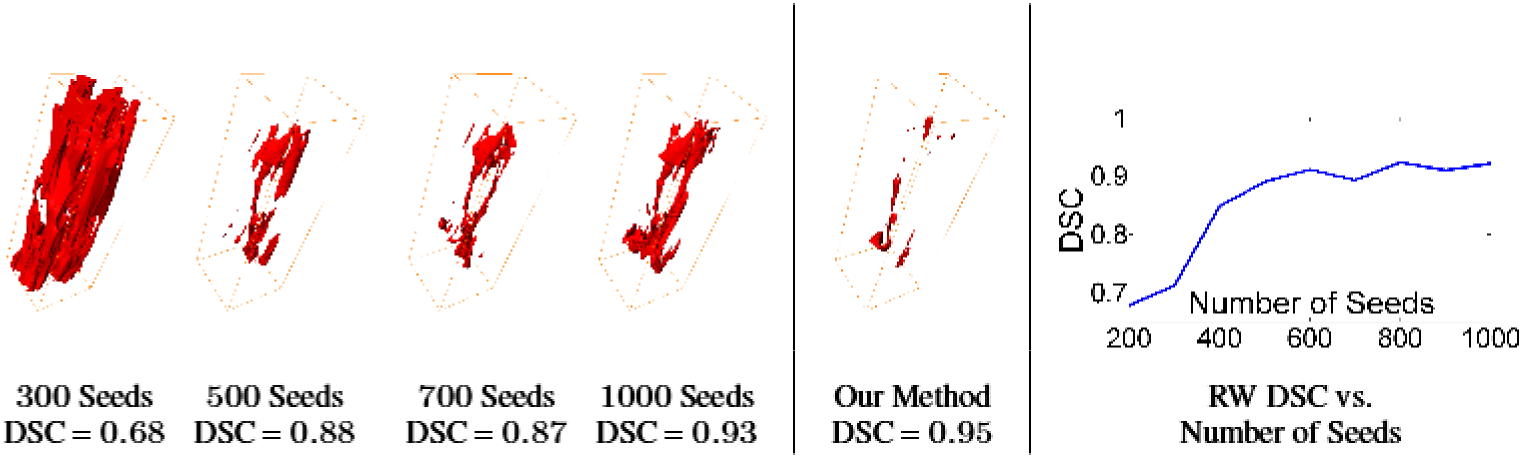
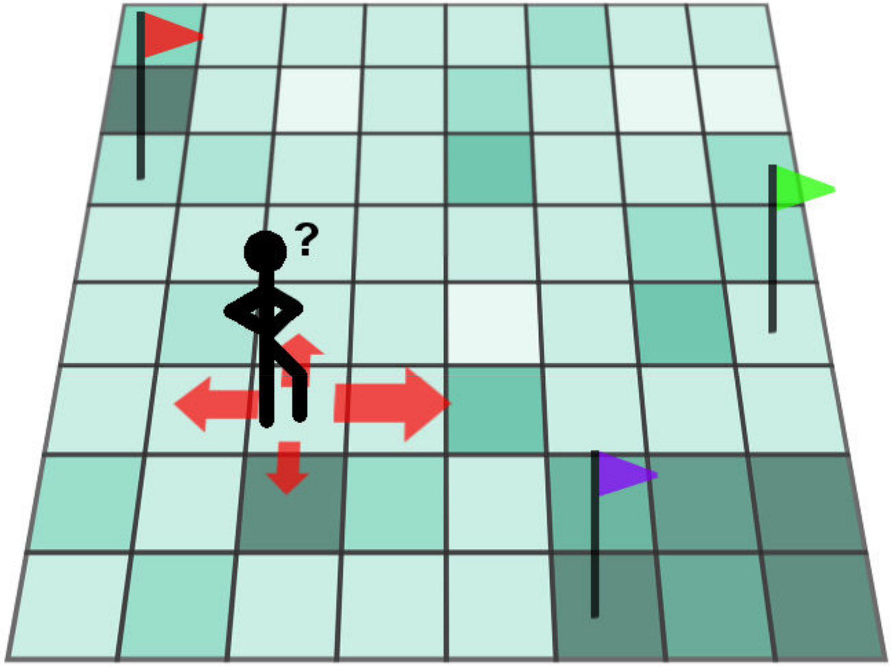


Fig. 6: The areas of mis-segmentation by the RW algorithm with varying numbers of randomly selected seeds (left) and our algorithm (center), along with a graph showing how the average DSC of RW changes with the number of seeds used (right).

Segmentation with random walks (Grady, 2006)

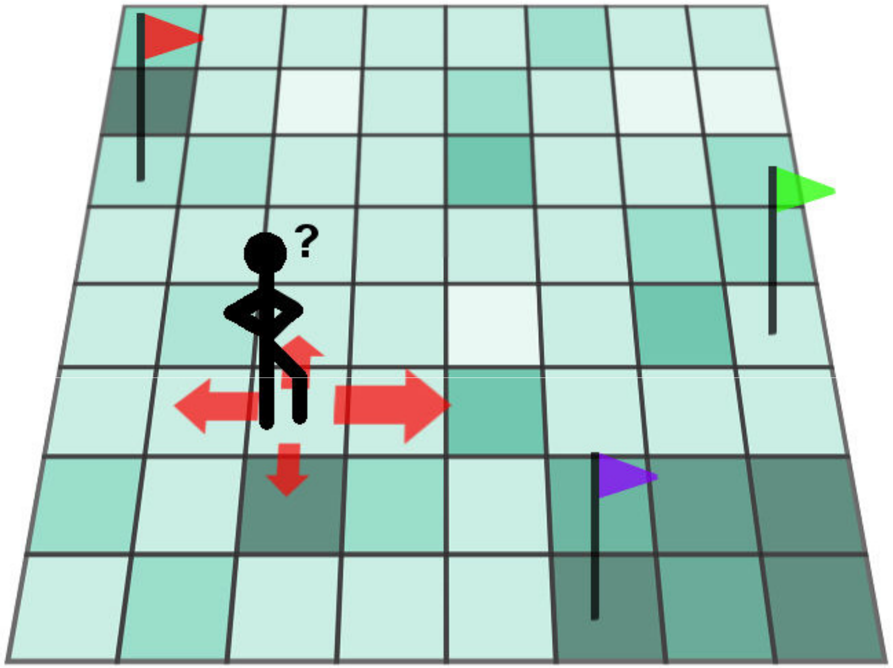


Let’s consider a walker that is moving randomly on the ground, to progress from one position, he had many possibilities.

The probability of moving to a specefic position is inversily proportionnel to the effort spent to make the step.

The walker will prefer easy path (flat) then more difficult ones (bumpy ground)

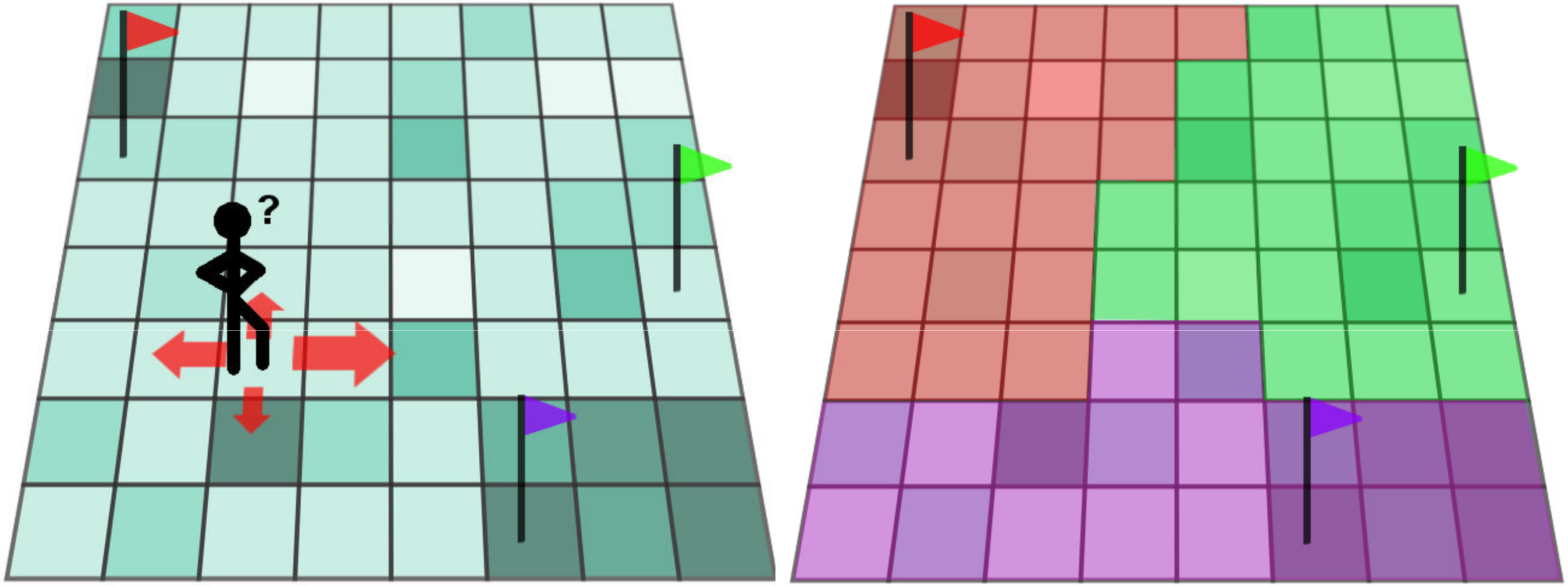
Segmentation with random walks (Grady, 2006)



Assume that there are some landmarks on the ground.

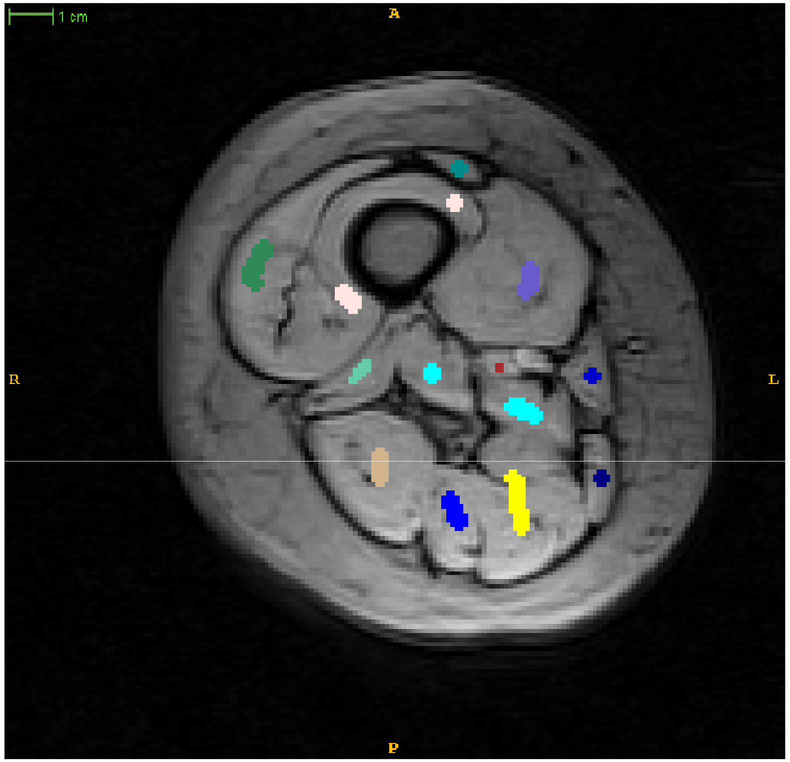
Q: What is the probability that the random walks arrives first to the green/red/ purple landmark ?

Segmentation with random walks (Grady, 2006)

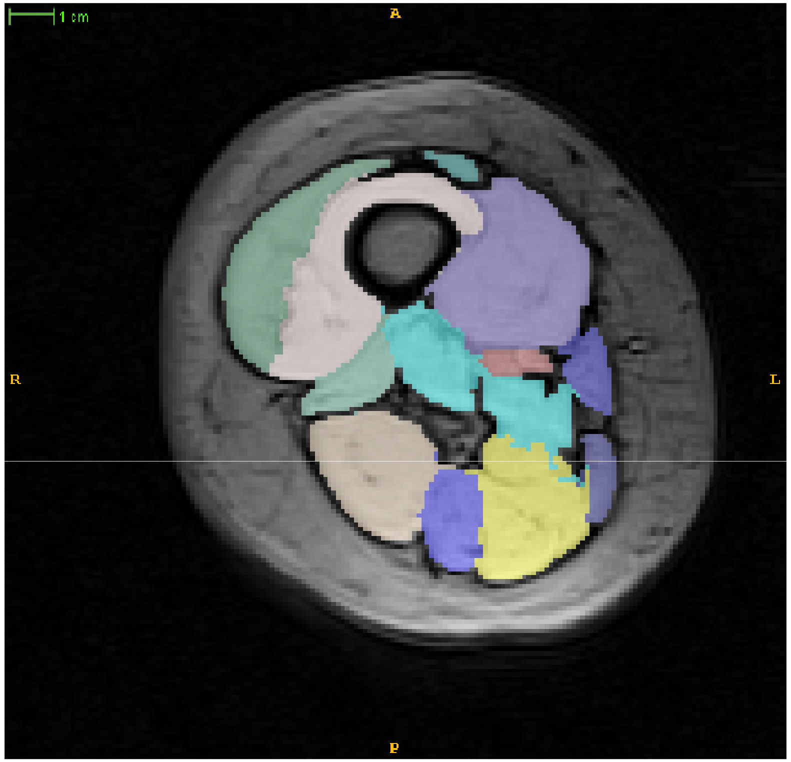


A : The random walk algorithms

Segmentation with random walks (Grady, 2006)



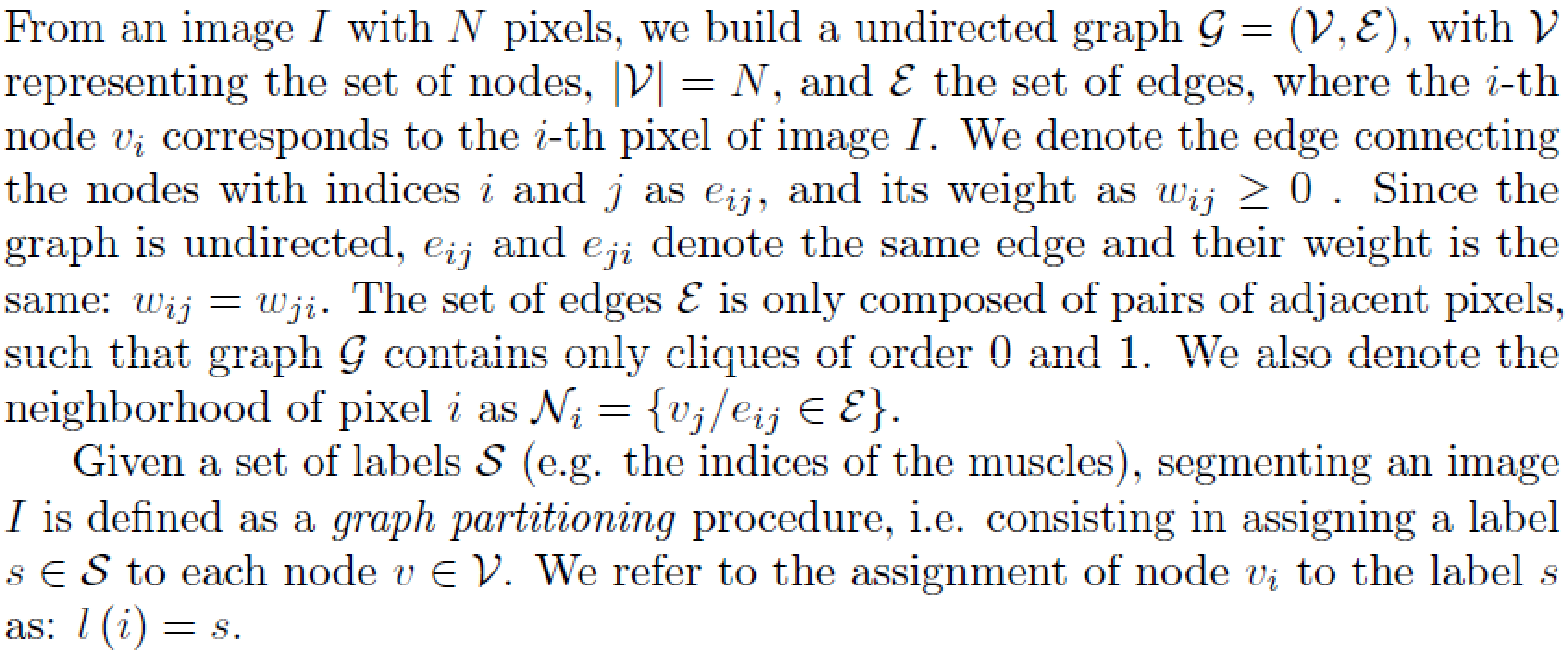
Non segmented view with **drawn** seeds



automatic segmentation

Segmentation with random walks (Grady, 2006)

Graph based formulation

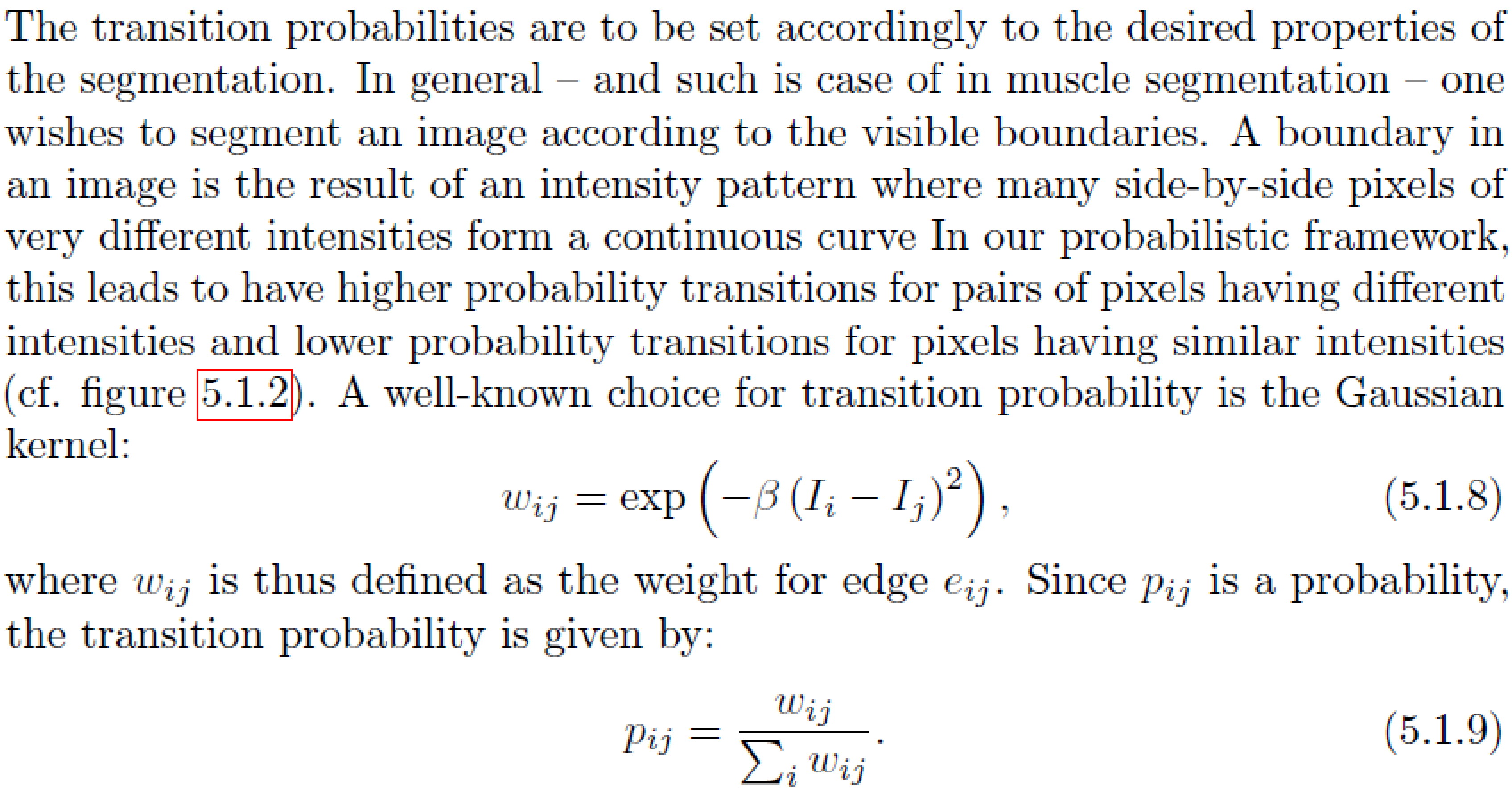


From an image I with N pixels, we build a undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $\mathcal{V}$ representing the set of nodes, $|\mathcal{V}| = N$, and $\mathcal{E}$ the set of edges, where the i-th node v_i corresponds to the i-th pixel of image I. We denote the edge connecting the nodes with indices i and j as e_{ij}, and its weight as $w_{ij} \geq 0$. Since the graph is undirected, e_{ij} and e_{ji} denote the same edge and their weight is the same: $w_{ij} = w_{ji}$. The set of edges $\mathcal{E}$ is only composed of pairs of adjacent pixels, such that graph $\mathcal{G}$ contains only cliques of order 0 and 1. We also denote the neighborhood of pixel i as $\mathcal{N}_i = \{v_j / e_{ij} \in \mathcal{E}\}$.

Given a set of labels $\mathcal{S}$ (e.g. the indices of the muscles), segmenting an image I is defined as a *graph partitioning* procedure, i.e. consisting in assigning a label $s \in \mathcal{S}$ to each node $v \in \mathcal{V}$. We refer to the assignment of node v_i to the label s as: $l(i) = s$.

Segmentation with random walks (Grady, 2006)

Transition probabilities definition



The transition probabilities are to be set accordingly to the desired properties of the segmentation. In general – and such is case of in muscle segmentation – one wishes to segment an image according to the visible boundaries. A boundary in an image is the result of an intensity pattern where many side-by-side pixels of very different intensities form a continuous curve In our probabilistic framework, this leads to have higher probability transitions for pairs of pixels having different intensities and lower probability transitions for pixels having similar intensities (cf. figure 5.1.2). A well-known choice for transition probability is the Gaussian kernel:

$$w_{ij} = \exp\left(-\beta\left(I_i - I_j\right)^2\right), \tag{5.1.8}$$

where w_{ij} is thus defined as the weight for edge e_{ij}. Since p_{ij} is a probability, the transition probability is given by:

$$p_{ij} = \frac{w_{ij}}{\sum_i w_{ij}}. \tag{5.1.9}$$

Segmentation with random walks (Grady, 2006)

Labelling

In this framework, the global label assignment is modeled by a random variable. We denote the probability of assignment of voxel i to label s as $x_i^s = \Pr\left(l\left(i\right) = s\right)$. In this notation, $l\left(i\right) = \arg\max_s x_i^s$. We denote the transition probability from node i to node j as p_{ij}. In general, we have $p_{ij} \neq p_{ji}$.

Assume we possess a set $\mathcal{V}_M$ of seeds, i.e. pre-labeled nodes *for each label* (typically, manually marked voxels). We denote the set of unknown nodes as $\mathcal{V}_U$, such that $\mathcal{V}_M \cup \mathcal{V}_U = \mathcal{V}$ and $\mathcal{V}_M \cap \mathcal{V}_U = \emptyset$. For conveniency in the notations, we denote the sets of indices of the nodes in $\mathcal{V}_U$ and $\mathcal{V}_M$ as U and M respectively. Since the label of marked voxels is known for certain, their assignment probability is either 1 or 0

$$\forall i \in M,\ x_i^s = \begin{cases} 1 & l\left(i\right) = s, \\ 0 & l\left(i\right) \neq s. \end{cases} \tag{5.1.1}$$

We denote to the probability vector for label s as $\mathbf{x}^s$, where $\mathbf{x}^s$ contains the x_i^s for each voxel i. Then, without loss of generality, we can assume the variables in $\mathbf{x}^s$ are ordered so that we can write:

$$\mathbf{x}^s = \left[\begin{array}{c} \mathbf{x}_U^s \\ \mathbf{x}_M^s \end{array} \right], \tag{5.1.2}$$

Segmentation with random walks (Grady, 2006)

Objective function :

$$\forall v_i \in \mathcal{V}_U,\ x_i^s = \sum_{v_j \in \mathcal{N}_i} p_{ij} x_j^s .$$

$$E_{\mathrm{RW}}^s\left(\mathbf{x}^s\right) = \sum_{e_{ij} \in \mathcal{E}} w_{ij} \left(x_i^s - x_j^s\right)^2 ,$$

Let us define A, the non-normalized transition matrix – also designated as the *affinity matrix*:

$$A_{ij} = \begin{cases} w_{ij} & \textit{if } e_{ij} \in \mathcal{E}. \\ 0 & \text{otherwise,} \end{cases} \tag{5.1.10}$$

and D, the diagonal matrix such that:

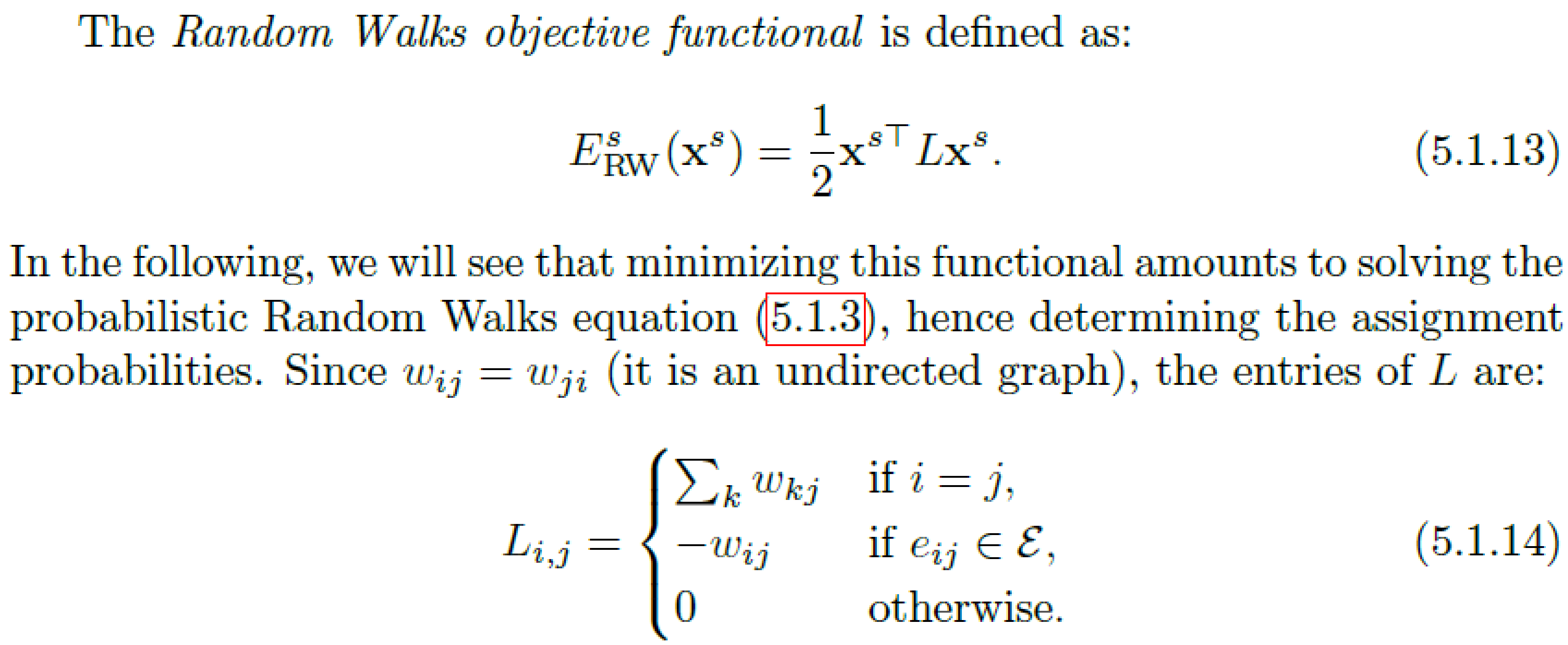
$$D_{ii} = \sum_j w_{ij}, \tag{5.1.11}$$

and L, the *unnormalized combinatorial Laplacian matrix*

$$L = D - A. \tag{5.1.12}$$

Segmentation with random walks (Grady, 2006)

Objective function



The *Random Walks objective functional* is defined as:

$$E^s_{\text{RW}}(\mathbf{x}^s) = \frac{1}{2}\mathbf{x}^{s\top} L \mathbf{x}^s. \tag{5.1.13}$$

In the following, we will see that minimizing this functional amounts to solving the probabilistic Random Walks equation (5.1.3), hence determining the assignment probabilities. Since $w_{ij} = w_{ji}$ (it is an undirected graph), the entries of L are:

$$L_{i,j} = \begin{cases} \sum_k w_{kj} & \text{if } i = j, \\ -w_{ij} & \text{if } e_{ij} \in \mathcal{E}, \\ 0 & \text{otherwise.} \end{cases} \tag{5.1.14}$$

Segmentation with random walks (Grady, 2006)

Objective function

We decompose L into sub-blocks for marked and unknown nodes, hence defining subblocks L_U, L_M and B:

$$L = \begin{bmatrix} L_U & B \\ B^\top & L_M \end{bmatrix},$$

which allows us to rewrite equation (5.1.13) as:

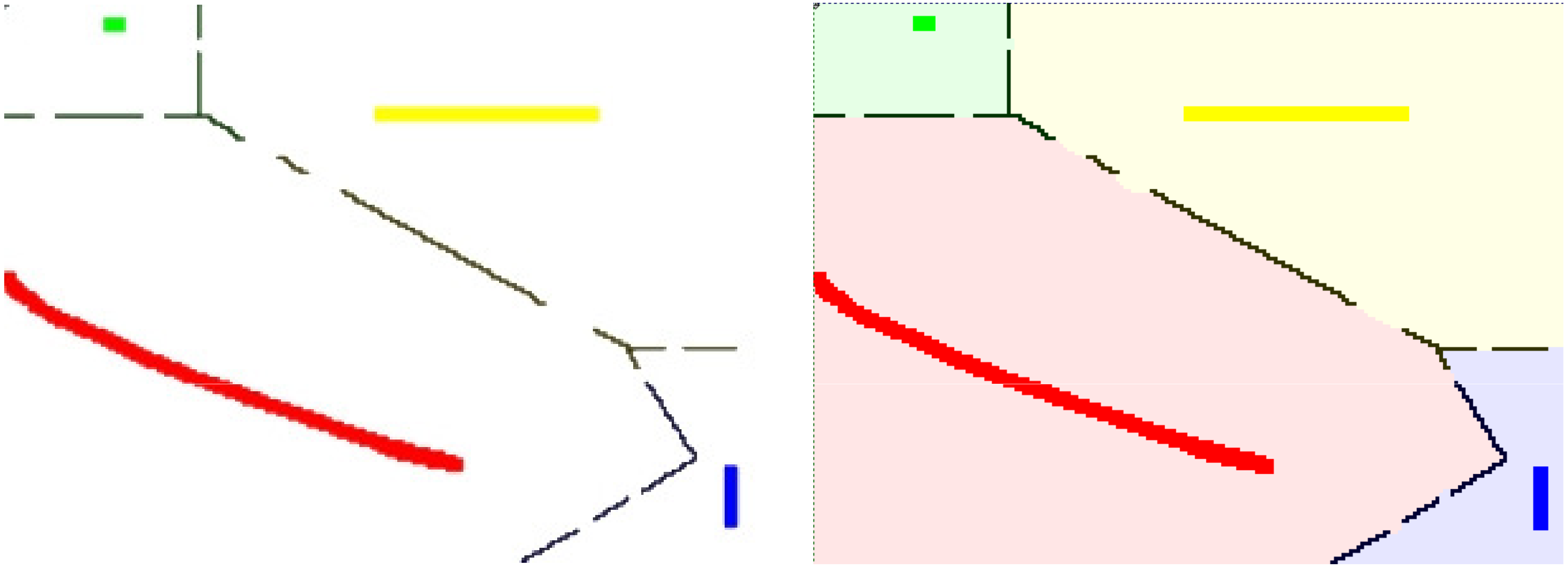
$$E_{\mathrm{RW}}(\mathbf{x}^s) = \frac{1}{2}\mathbf{x}_U^{s\top} L_U \mathbf{x}_U^s + \mathbf{x}_U^{s\top} B \mathbf{x}_M^s + \frac{1}{2}\mathbf{x}_M^{s\top} L_M \mathbf{x}_M^s . \tag{5.1.17}$$

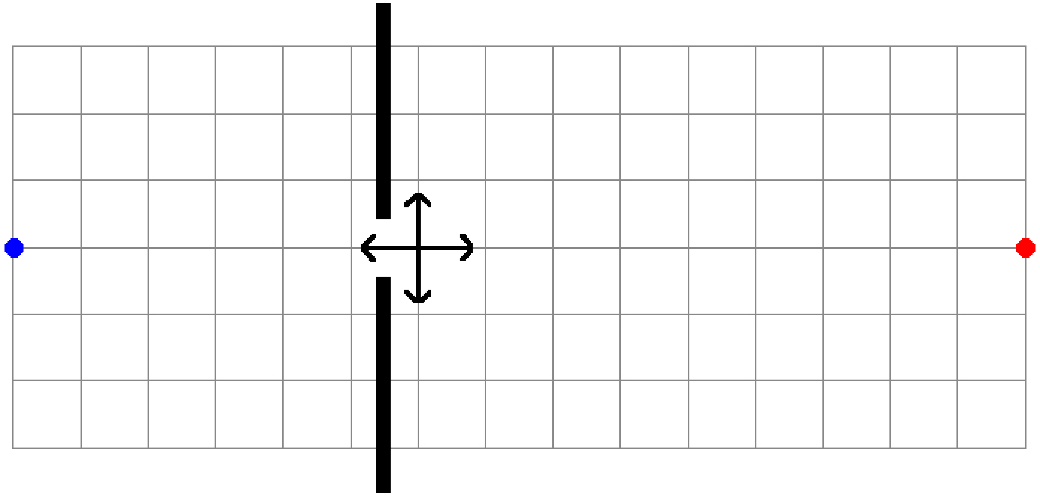
By differenciating (5.1.17) with respect to the unknown variables, we obtain:

$$L_U \mathbf{x}_U^s = -B \mathbf{x}_M^s . \tag{5.1.18}$$

Segmentation with random walks (Grady, 2006)

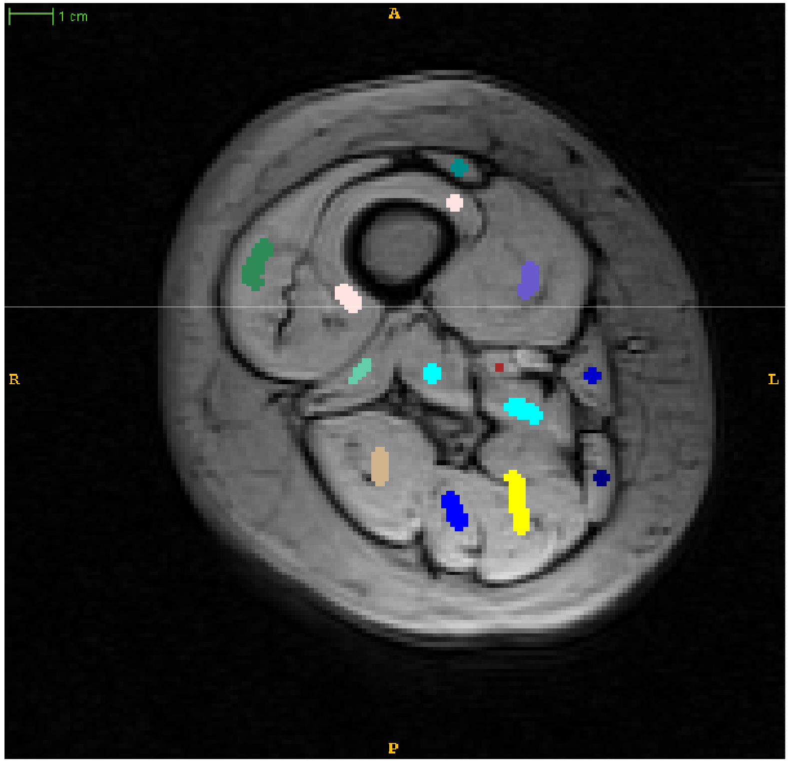
Results : it naturally respects weaks boundary



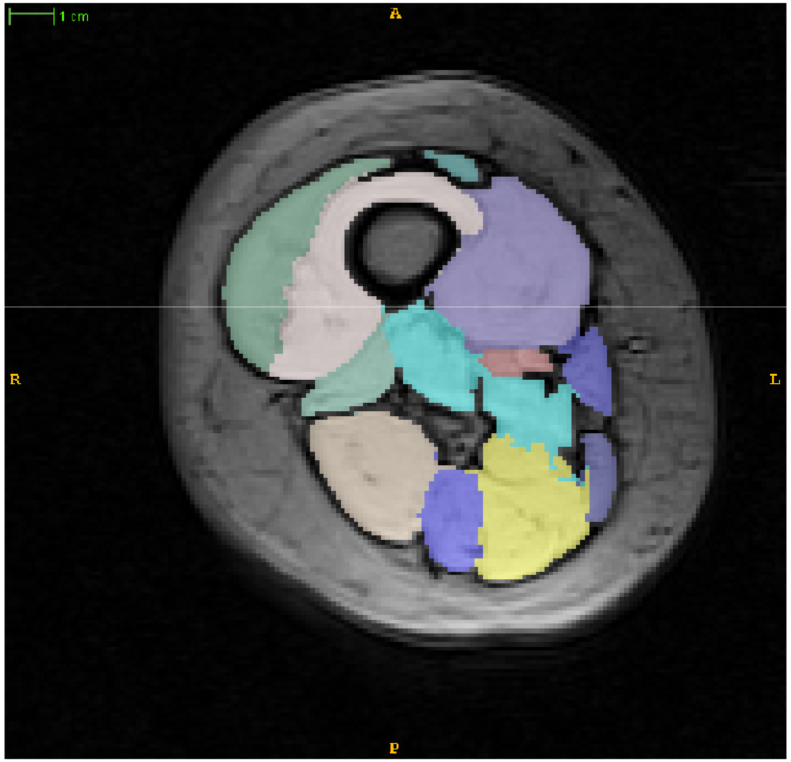


Segmentation with random walks (Grady, 2006)

Application to muscle segmentation



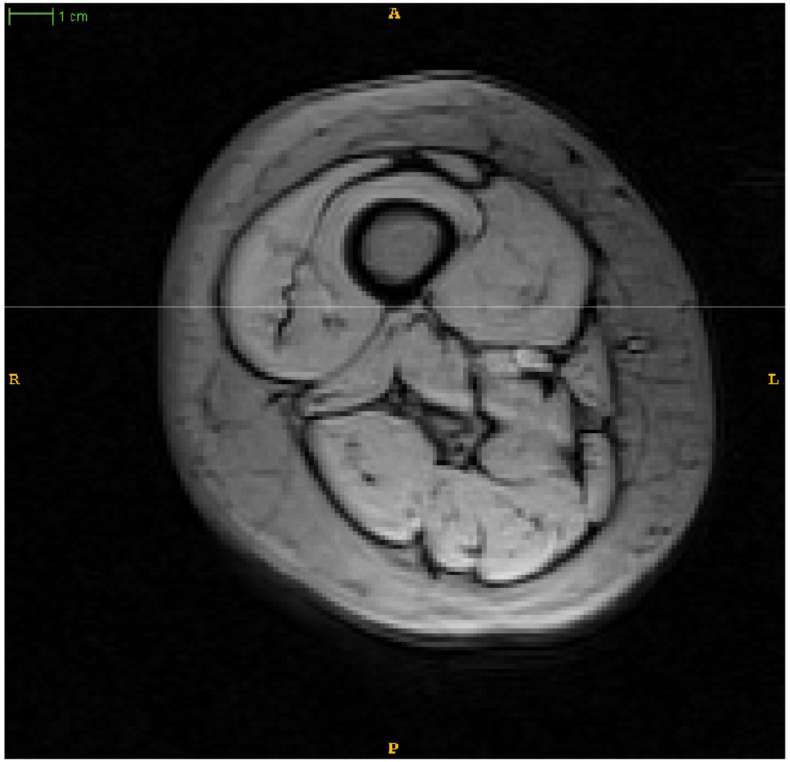
non segmented view with
drawn seeds



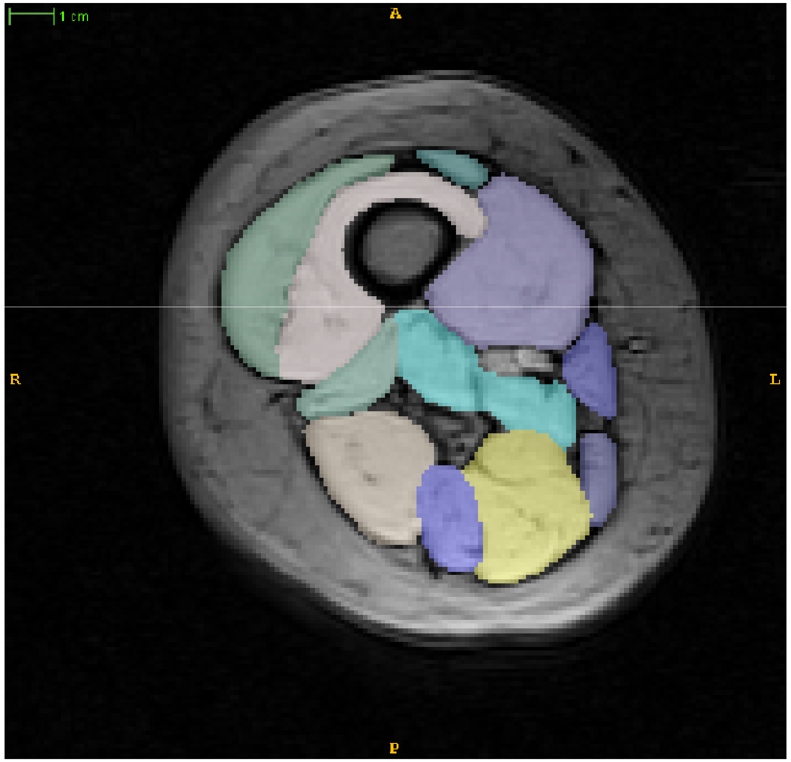
automatic segmentation

Segmentation with random walks (Grady, 2006)

Application to muscle segmentation



a non segmented view



manual segmentation

Segmentation with random walks (Grady, 2006)

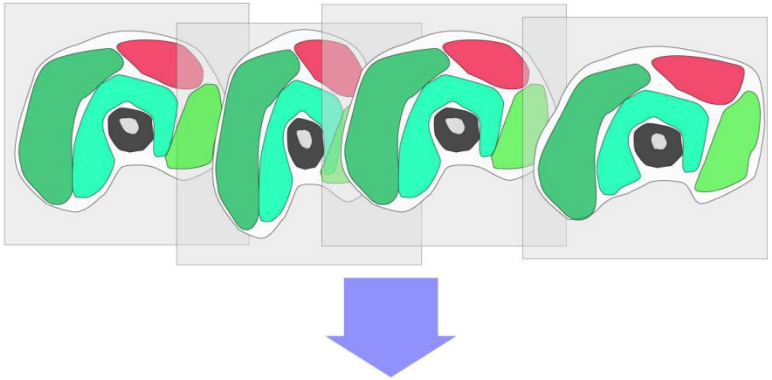
Application to muscle segmentation

- User interaction: drawing “seeds”
- 20 min instead of 4 hours
- Random Walk computation (< 5 min)
- Accuracy of the Random Walk segmentation: 90%

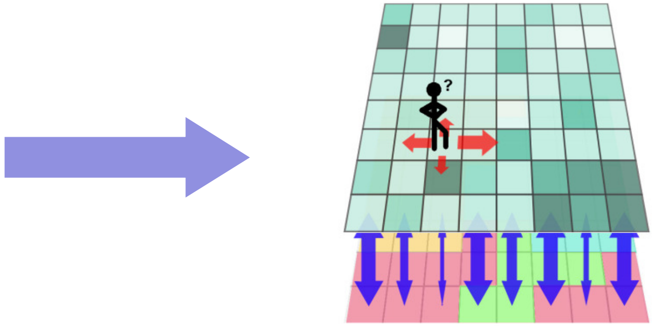
RW & Prior Shape Model (Baudin et al MICCAI 2012)

Objective function :

$$\forall v_i \in \mathcal{V},\; x_i^s = (1-\lambda_i) \sum_{v_j \in \mathcal{N}_i} p_{ij} x_j^s + \lambda_i x_{0\,i}^s,$$



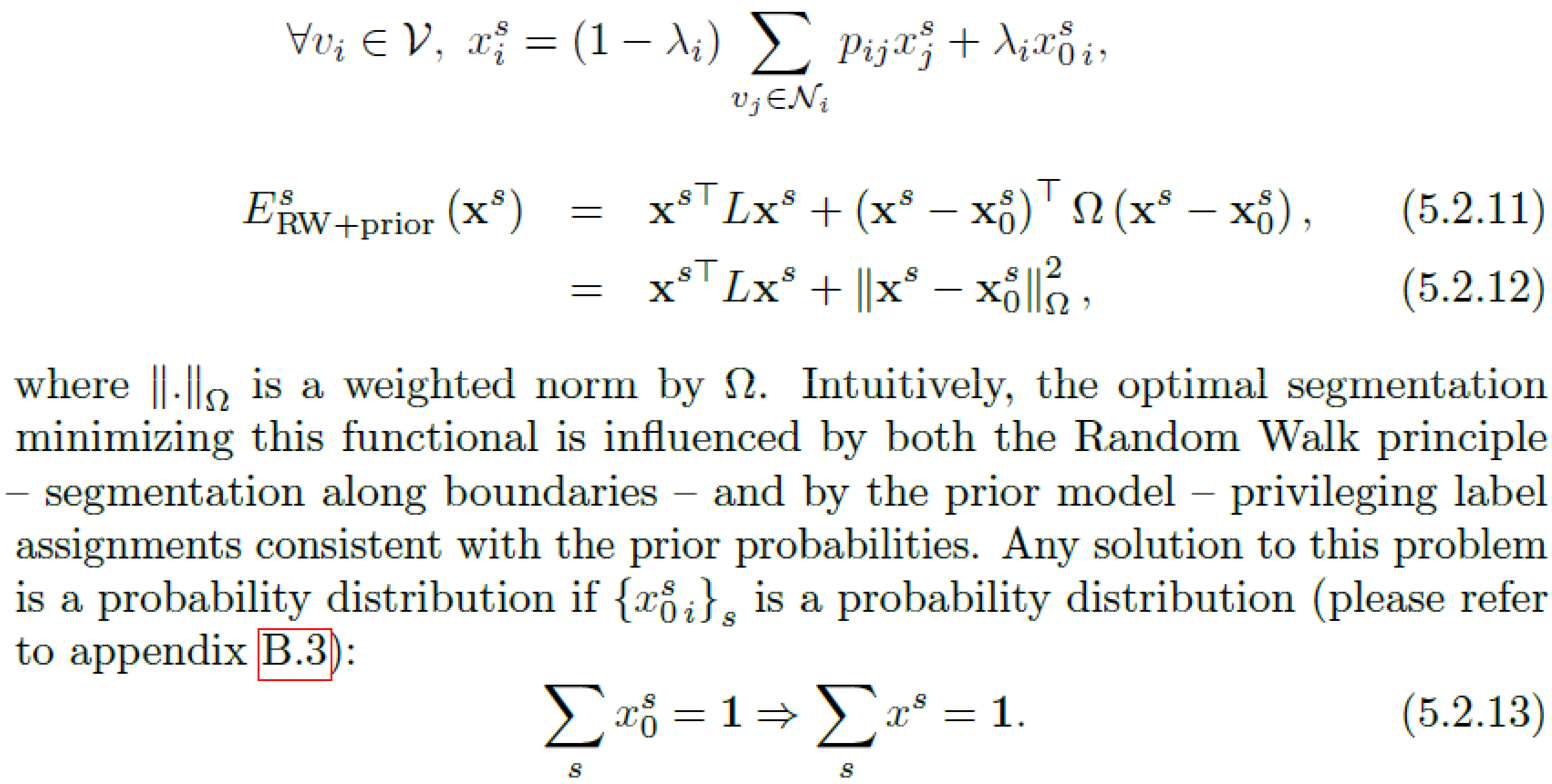
Mean shape



The random walker is guided by the muscle shape model instead of seeds.

RW & Prior Shape Model (Baudin et al MICCAI 2012)

Objective function :



$$\forall v_i \in \mathcal{V},\ x_i^s = (1-\lambda_i) \sum_{v_j \in \mathcal{N}_i} p_{ij} x_j^s + \lambda_i x_{0\,i}^s,$$

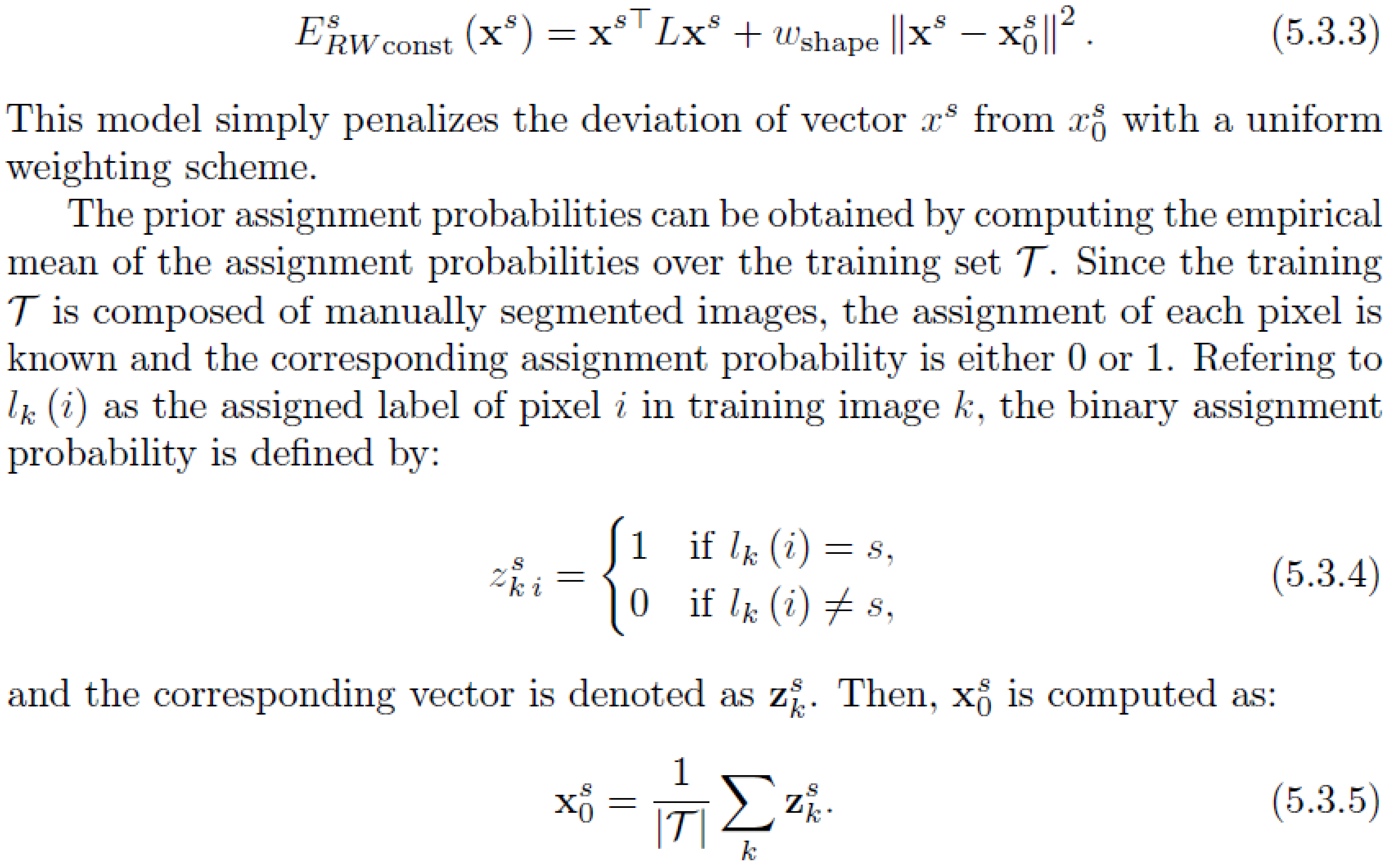
$$\begin{aligned} E^s_{\text{RW+prior}}\left(\mathbf{x}^s\right) &= \mathbf{x}^{s\top} L \mathbf{x}^s + \left(\mathbf{x}^s - \mathbf{x}_0^s\right)^\top \Omega \left(\mathbf{x}^s - \mathbf{x}_0^s\right), && (5.2.11) \\ &= \mathbf{x}^{s\top} L \mathbf{x}^s + \|\mathbf{x}^s - \mathbf{x}_0^s\|^2_\Omega, && (5.2.12) \end{aligned}$$

where $\|.\|_\Omega$ is a weighted norm by Ω. Intuitively, the optimal segmentation minimizing this functional is influenced by both the Random Walk principle – segmentation along boundaries – and by the prior model – privileging label assignments consistent with the prior probabilities. Any solution to this problem is a probability distribution if $\{x_{0\,i}^s\}_s$ is a probability distribution (please refer to appendix B.3):

$$\sum_s x_0^s = 1 \Rightarrow \sum_s x^s = 1. \qquad (5.2.13)$$

RW & Prior Shape Model (Baudin et al MICCAI 2012)

Weight s of the prior knowledge : constant weight



$$E^s_{RW\text{const}}\left(\mathbf{x}^s\right) = \mathbf{x}^{s\top} L \mathbf{x}^s + w_{\text{shape}} \left\|\mathbf{x}^s - \mathbf{x}^s_0\right\|^2 . \qquad (5.3.3)$$

This model simply penalizes the deviation of vector x^s from x^s_0 with a uniform weighting scheme.

The prior assignment probabilities can be obtained by computing the empirical mean of the assignment probabilities over the training set $\mathcal{T}$. Since the training $\mathcal{T}$ is composed of manually segmented images, the assignment of each pixel is known and the corresponding assignment probability is either 0 or 1. Refering to $l_k\left(i\right)$ as the assigned label of pixel i in training image k, the binary assignment probability is defined by:

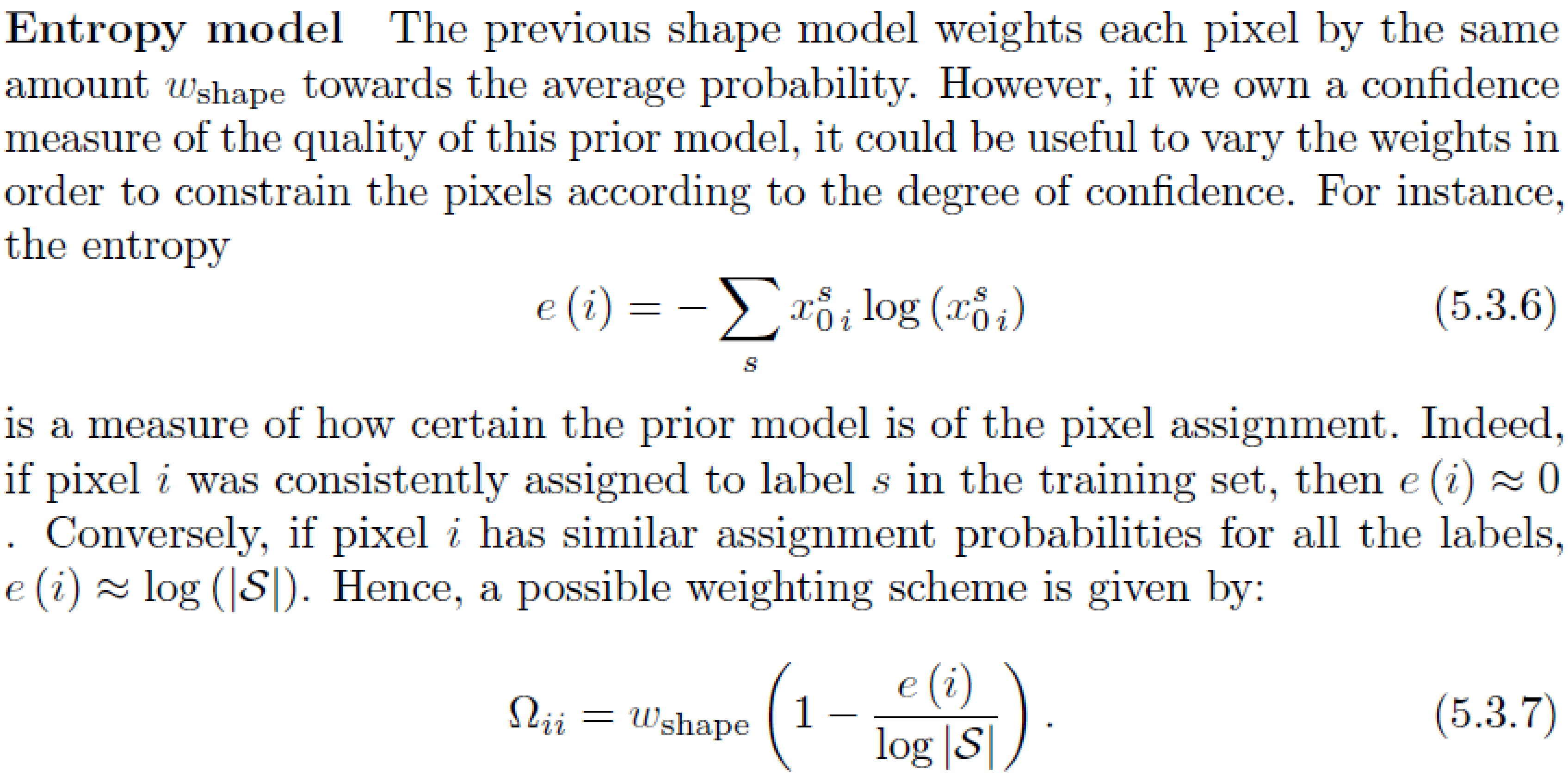
$$z^s_{k\,i} = \begin{cases} 1 & \text{if } l_k\left(i\right) = s, \\ 0 & \text{if } l_k\left(i\right) \neq s, \end{cases} \qquad (5.3.4)$$

and the corresponding vector is denoted as $\mathbf{z}^s_k$. Then, $\mathbf{x}^s_0$ is computed as:

$$\mathbf{x}^s_0 = \frac{1}{|\mathcal{T}|} \sum_k \mathbf{z}^s_k . \qquad (5.3.5)$$

RW & Prior Shape Model (Baudin et al MICCAI 2012)

Weight s of the prior knowledge : Entropy



Entropy model The previous shape model weights each pixel by the same amount w_{shape} towards the average probability. However, if we own a confidence measure of the quality of this prior model, it could be useful to vary the weights in order to constrain the pixels according to the degree of confidence. For instance, the entropy

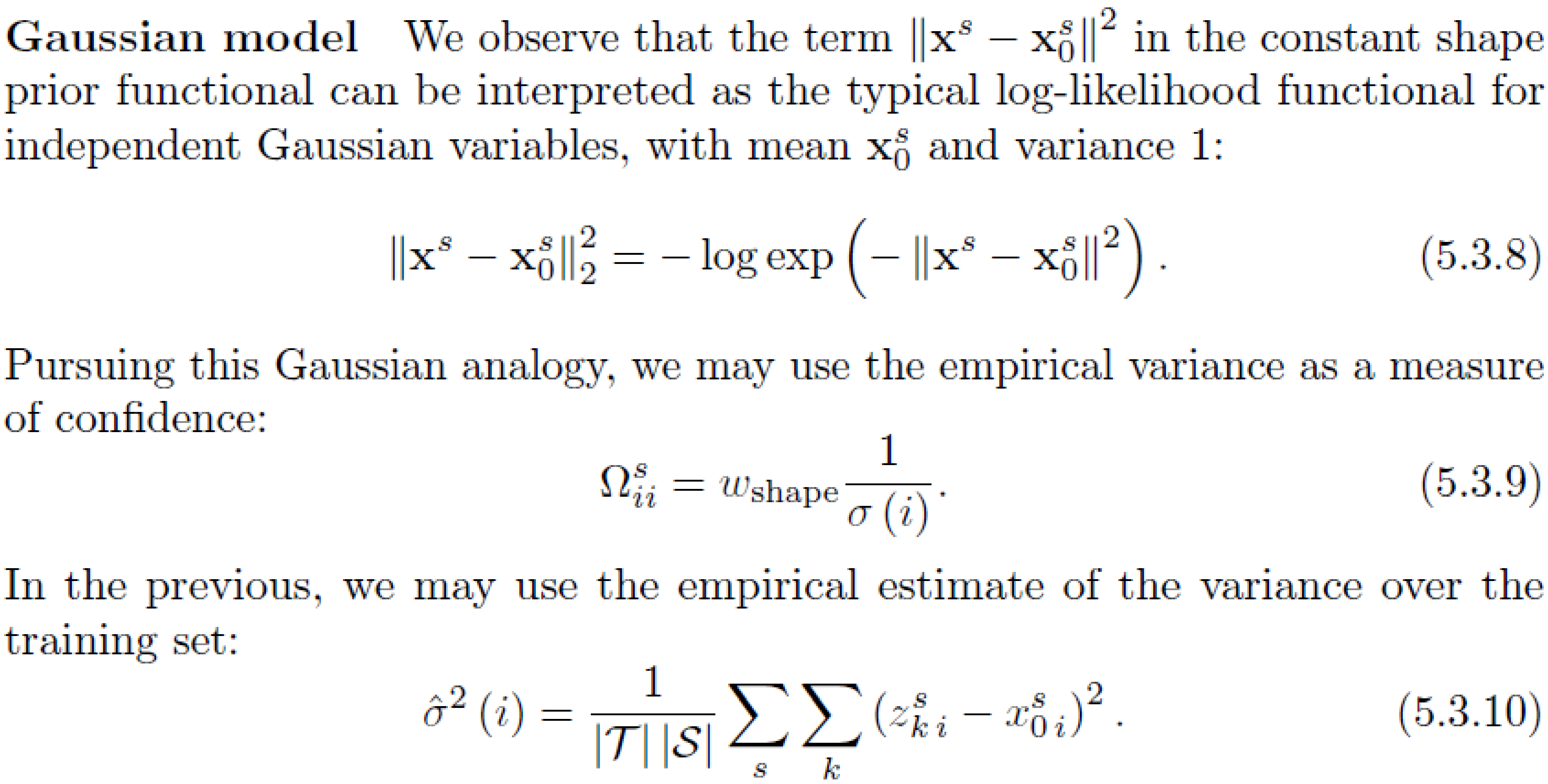
$$e\left(i\right) = -\sum_{s} x_{0\,i}^{s} \log\left(x_{0\,i}^{s}\right) \tag{5.3.6}$$

is a measure of how certain the prior model is of the pixel assignment. Indeed, if pixel i was consistently assigned to label s in the training set, then $e\left(i\right) \approx 0$. Conversely, if pixel i has similar assignment probabilities for all the labels, $e\left(i\right) \approx \log\left(\left|\mathcal{S}\right|\right)$. Hence, a possible weighting scheme is given by:

$$\Omega_{ii} = w_{\text{shape}} \left(1 - \frac{e\left(i\right)}{\log\left|\mathcal{S}\right|}\right) . \tag{5.3.7}$$

RW & Prior Shape Model (Baudin et al, MICCAI 2012)

Weight s of the prior knowledge : Gaussian weights



Gaussian model We observe that the term $\|\mathbf{x}^s - \mathbf{x}_0^s\|^2$ in the constant shape prior functional can be interpreted as the typical log-likelihood functional for independent Gaussian variables, with mean $\mathbf{x}_0^s$ and variance 1:

$$\|\mathbf{x}^s - \mathbf{x}_0^s\|_2^2 = -\log\exp\left(-\|\mathbf{x}^s - \mathbf{x}_0^s\|^2\right). \tag{5.3.8}$$

Pursuing this Gaussian analogy, we may use the empirical variance as a measure of confidence:

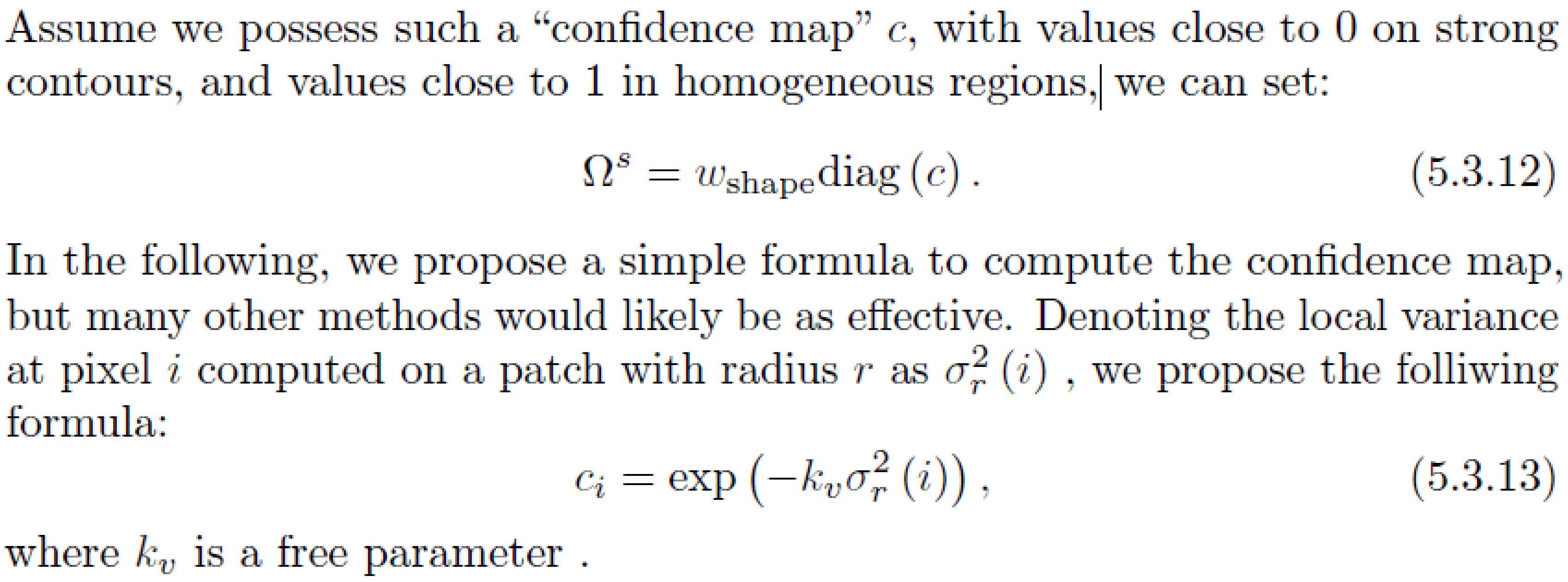
$$\Omega_{ii}^s = w_{\text{shape}} \frac{1}{\sigma(i)}. \tag{5.3.9}$$

In the previous, we may use the empirical estimate of the variance over the training set:

$$\hat{\sigma}^2(i) = \frac{1}{|\mathcal{T}|\,|\mathcal{S}|} \sum_s \sum_k \left(z_{k\,i}^s - x_{0\,i}^s\right)^2. \tag{5.3.10}$$

RW & Prior Shape Model (Baudin et al, MICCAI 2012)

Weight s of the prior knowledge : Confidence map



Assume we possess such a “confidence map” c, with values close to 0 on strong contours, and values close to 1 in homogeneous regions, we can set:

$$\Omega^s = w_{\text{shape}} \text{diag}\left(c\right) . \tag{5.3.12}$$

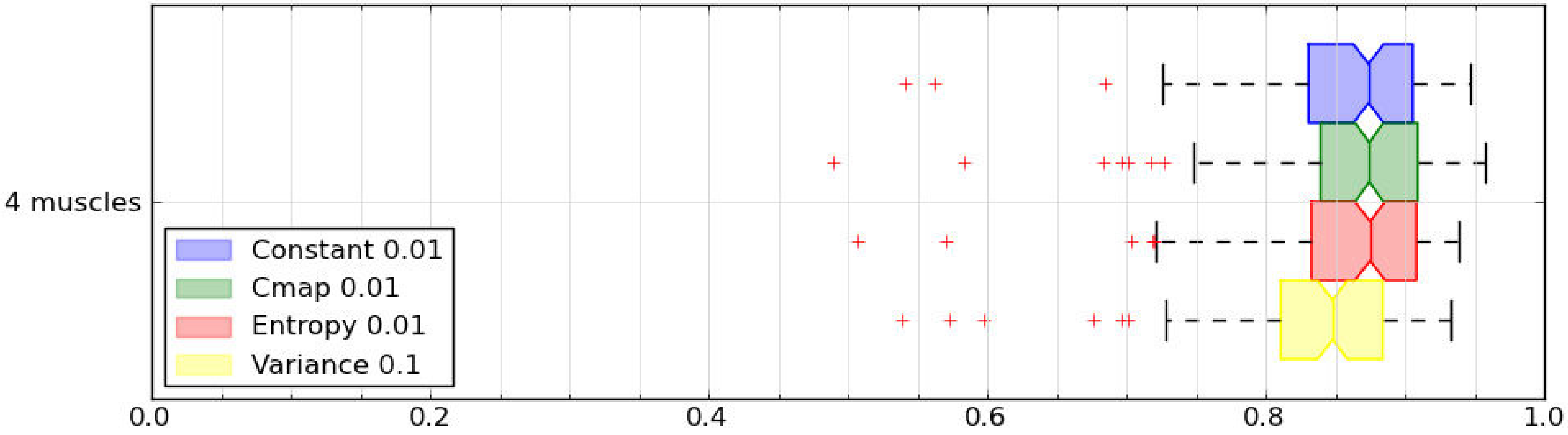
In the following, we propose a simple formula to compute the confidence map, but many other methods would likely be as effective. Denoting the local variance at pixel i computed on a patch with radius r as $\sigma_r^2\left(i\right)$, we propose the folliwing formula:

$$c_i = \exp\left(-k_v \sigma_r^2\left(i\right)\right) , \tag{5.3.13}$$

where k_v is a free parameter .

RW & Prior Shape Model (Baudin et al, MICCAI 2012)

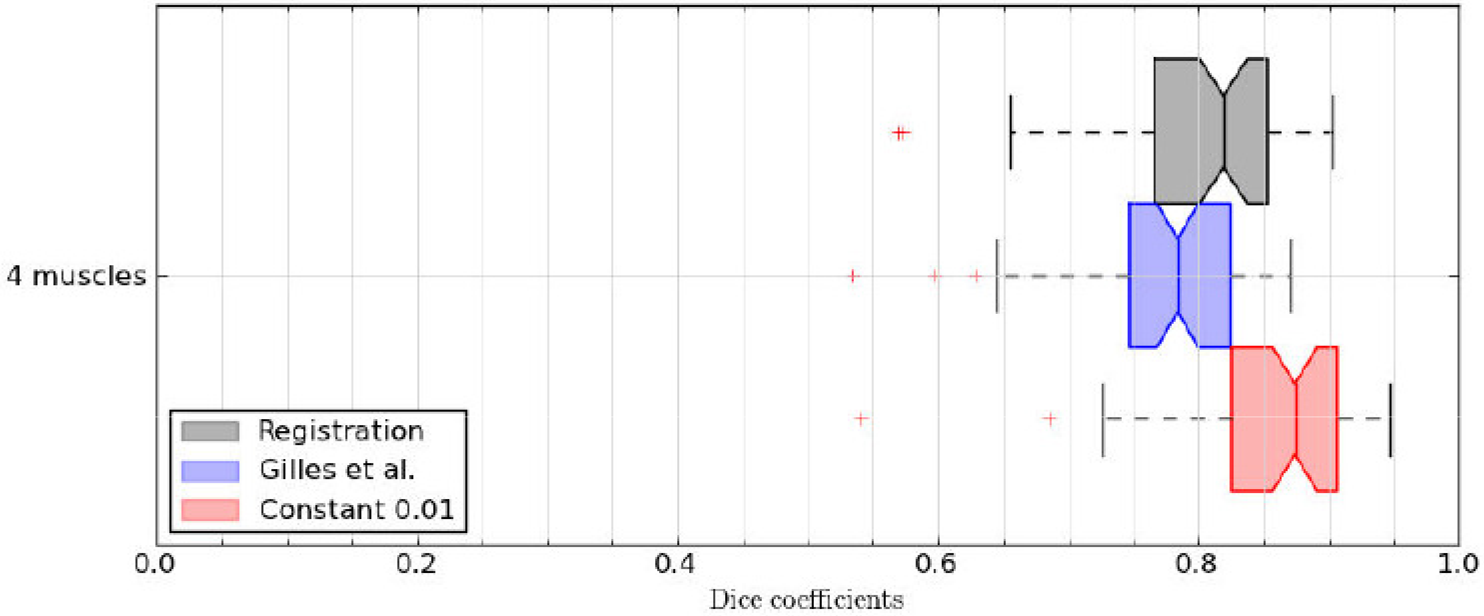
Experimental results



Comparing performances for various weighting schemes. the boxplots are generated from Dice coefficients for all labels (4 muscle database).

RW & Prior Shape Model (Baudin et al, MICCAI 2012)

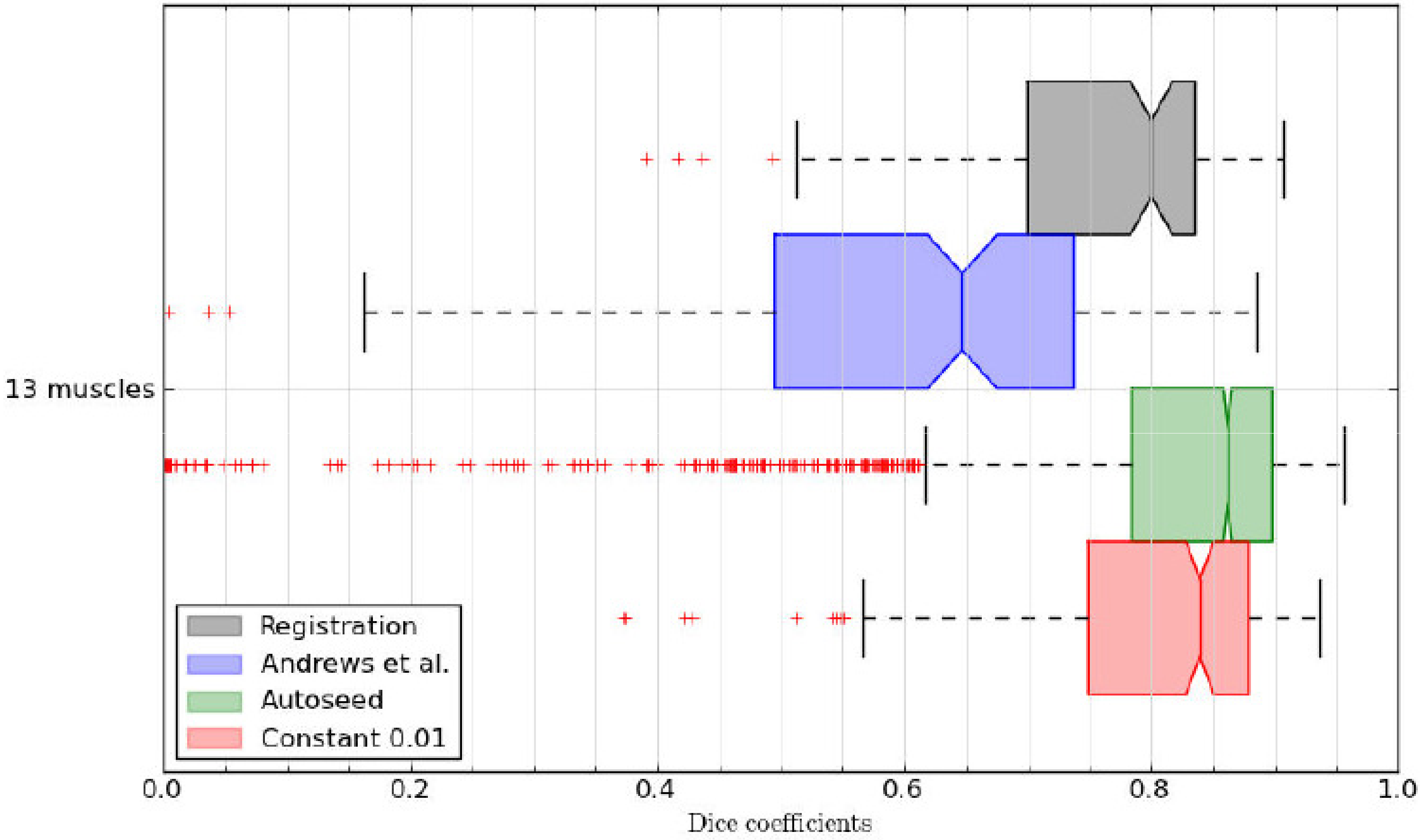
Experimental results



Comparison of our method (bottom box) with the “registration” method (top box) and the method from (Gilles and Pai, 2008) (middle box).
Dataset : subset of the 4 muscle dataset with only 15 test volumes.

RW & Prior Shape Model (Baudin et al, MICCAI 2012)

Experimental results



RW & Prior Shape Model (Baudin et al, MICCAI 2012)

Experimental results

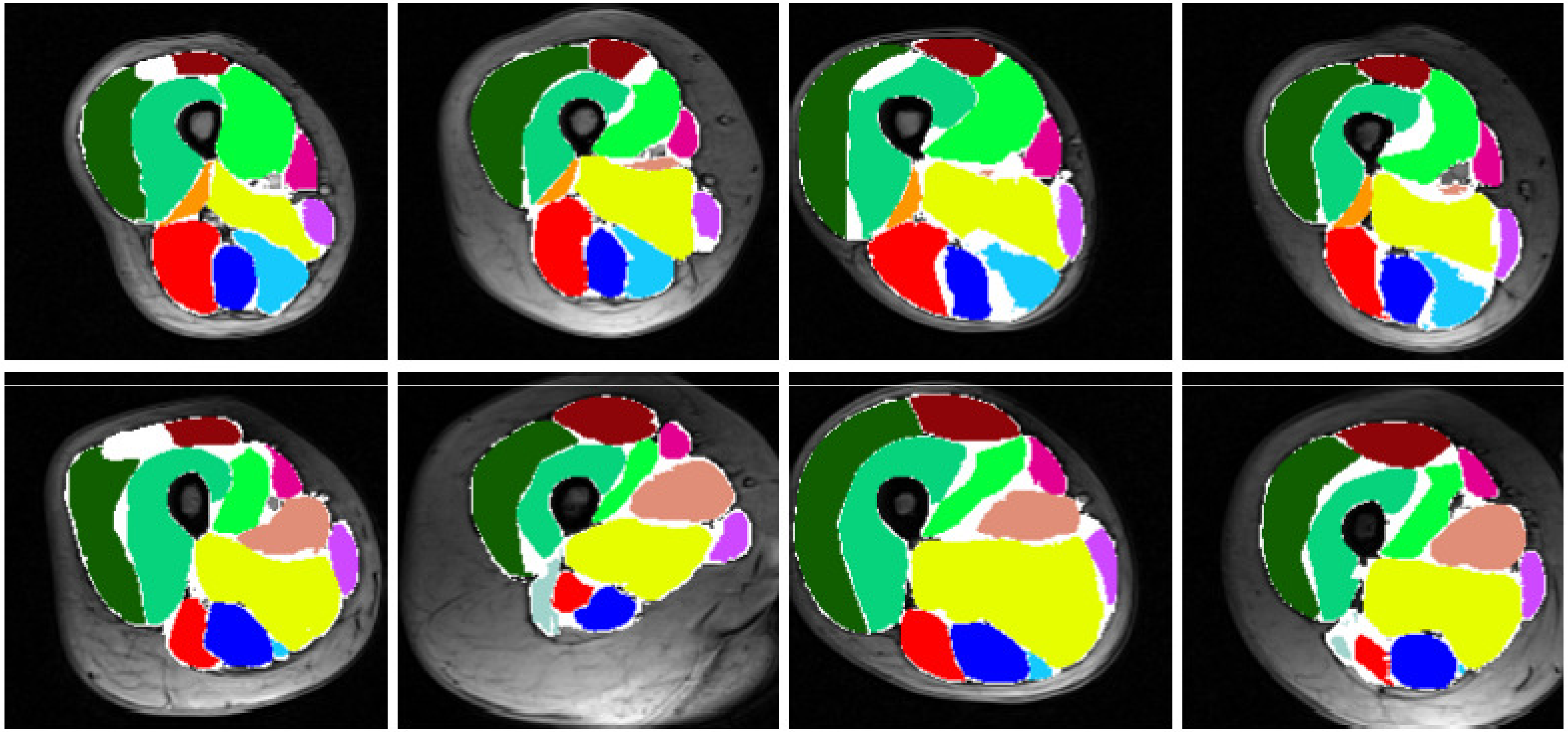
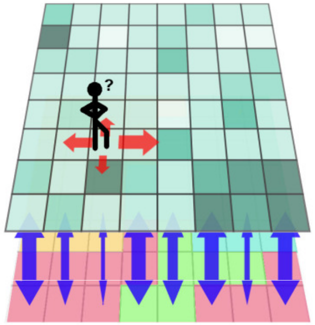


Fig. 3. Segmentation results obtained with the RW algorithm with shape prior and confidence map. Segmentation errors are shown in white.

RW & Prior Shape Model (Baudin et al, BMVC 2012)

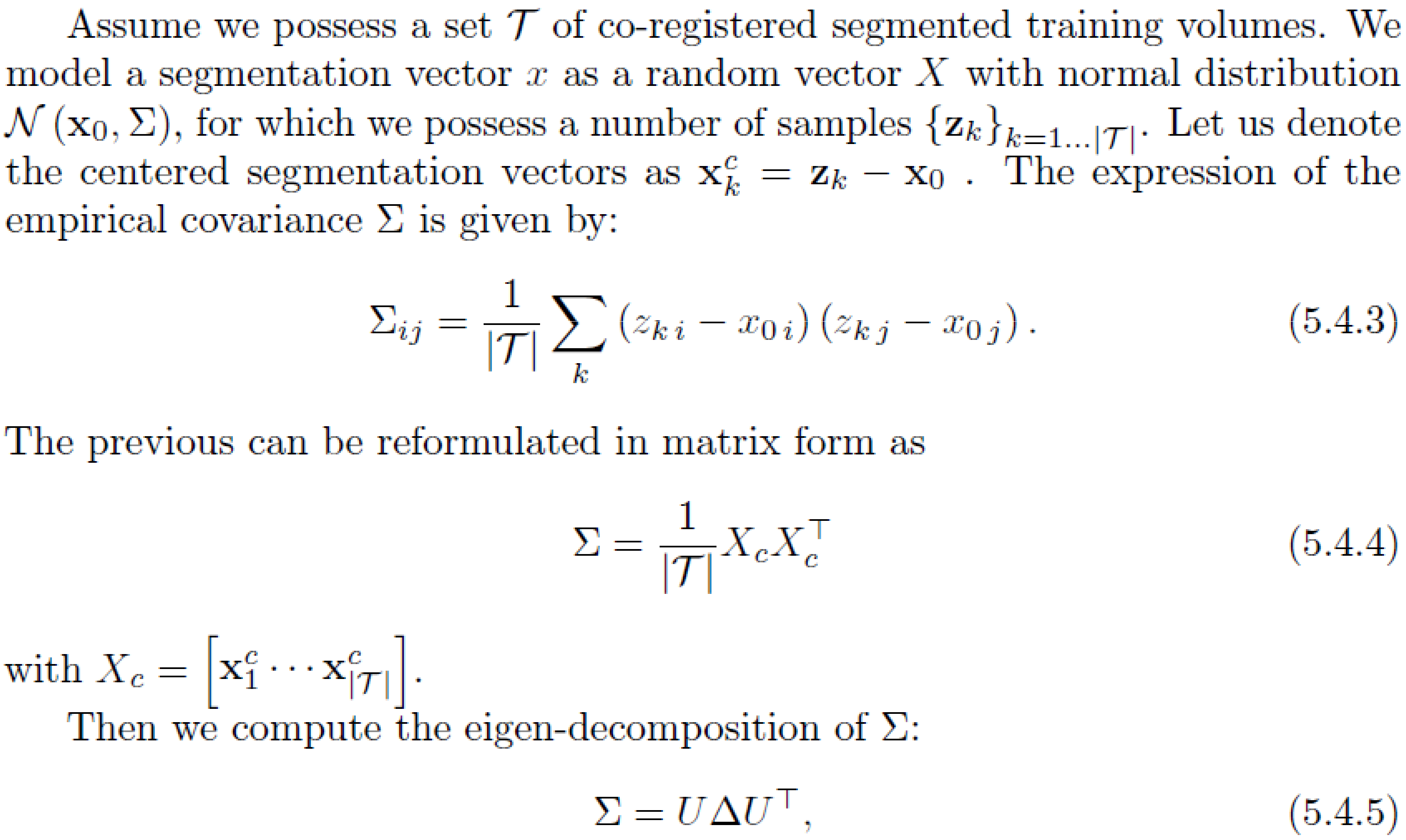
Mean shape and principal variation mode



The random walker is guided by the muscle shape model instead of seeds.

RW & Prior Shape Model (Baudin et al, BMVC 2012)

Model construction using PCA



Assume we possess a set $\mathcal{T}$ of co-registered segmented training volumes. We model a segmentation vector x as a random vector X with normal distribution $\mathcal{N}(\mathbf{x}_0, \Sigma)$, for which we possess a number of samples $\{\mathbf{z}_k\}_{k=1\ldots|\mathcal{T}|}$. Let us denote the centered segmentation vectors as $\mathbf{x}_k^c = \mathbf{z}_k - \mathbf{x}_0$. The expression of the empirical covariance Σ is given by:

$$\Sigma_{ij} = \frac{1}{|\mathcal{T}|} \sum_k \left(z_{k\,i} - x_{0\,i}\right)\left(z_{k\,j} - x_{0\,j}\right). \tag{5.4.3}$$

The previous can be reformulated in matrix form as

$$\Sigma = \frac{1}{|\mathcal{T}|} X_c X_c^\top \tag{5.4.4}$$

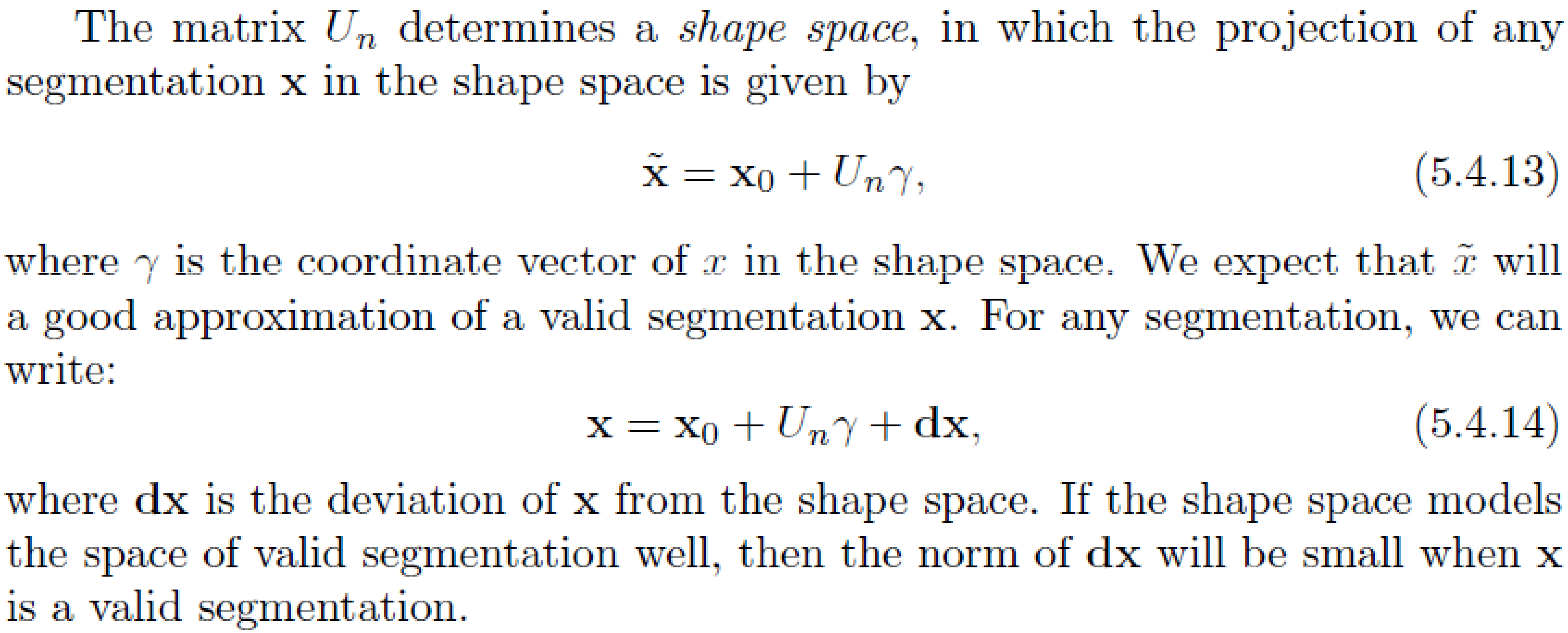
with $X_c = \left[\mathbf{x}_1^c \cdots \mathbf{x}_{|\mathcal{T}|}^c\right]$.

Then we compute the eigen-decomposition of Σ:

$$\Sigma = U \Delta U^\top, \tag{5.4.5}$$

RW & Prior Shape Model (Baudin et al, BMVC 2012)

Model construction using PCA



The matrix U_n determines a *shape space*, in which the projection of any segmentation $\mathbf{x}$ in the shape space is given by

$$\tilde{\mathbf{x}} = \mathbf{x}_0 + U_n\gamma, \tag{5.4.13}$$

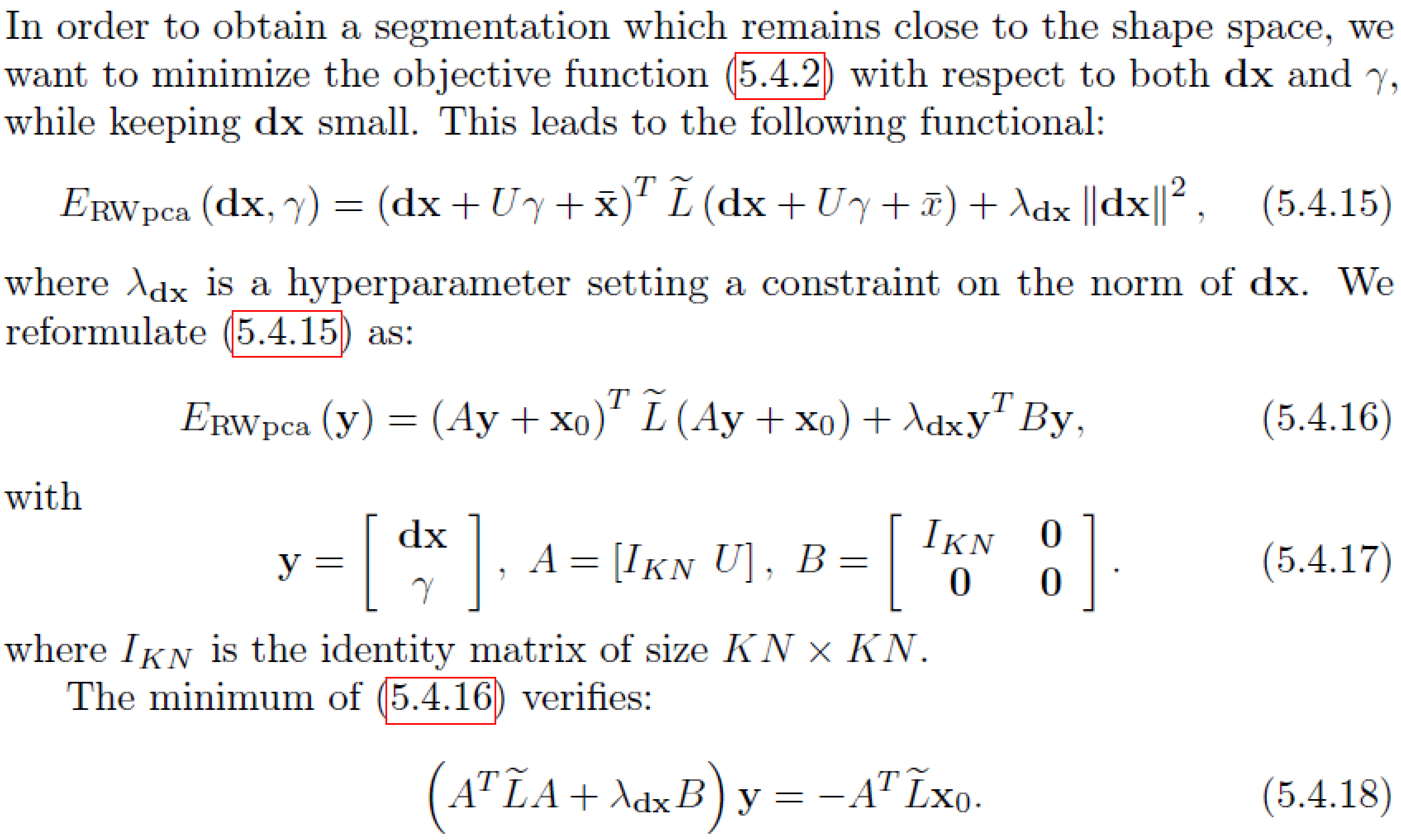
where γ is the coordinate vector of x in the shape space. We expect that $\tilde{x}$ will a good approximation of a valid segmentation $\mathbf{x}$. For any segmentation, we can write:

$$\mathbf{x} = \mathbf{x}_0 + U_n\gamma + \mathbf{dx}, \tag{5.4.14}$$

where $\mathbf{dx}$ is the deviation of $\mathbf{x}$ from the shape space. If the shape space models the space of valid segmentation well, then the norm of $\mathbf{dx}$ will be small when $\mathbf{x}$ is a valid segmentation.

RW & Prior Shape Model (Baudin et al, BMVC 2012)

Objective function



In order to obtain a segmentation which remains close to the shape space, we want to minimize the objective function (5.4.2) with respect to both $\mathbf{dx}$ and γ, while keeping $\mathbf{dx}$ small. This leads to the following functional:

$$E_{\mathrm{RWpca}}\left(\mathbf{dx},\gamma\right) = \left(\mathbf{dx} + U\gamma + \bar{\mathbf{x}}\right)^T \widetilde{L}\left(\mathbf{dx} + U\gamma + \bar{x}\right) + \lambda_{\mathbf{dx}}\left\|\mathbf{dx}\right\|^2, \qquad (5.4.15)$$

where $\lambda_{\mathbf{dx}}$ is a hyperparameter setting a constraint on the norm of $\mathbf{dx}$. We reformulate (5.4.15) as:

$$E_{\mathrm{RWpca}}\left(\mathbf{y}\right) = \left(A\mathbf{y} + \mathbf{x}_0\right)^T \widetilde{L}\left(A\mathbf{y} + \mathbf{x}_0\right) + \lambda_{\mathbf{dx}}\mathbf{y}^T B\mathbf{y}, \qquad (5.4.16)$$

with

$$\mathbf{y} = \begin{bmatrix} \mathbf{dx} \\ \gamma \end{bmatrix}, \; A = \left[I_{KN} \; U\right], \; B = \begin{bmatrix} I_{KN} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}. \qquad (5.4.17)$$

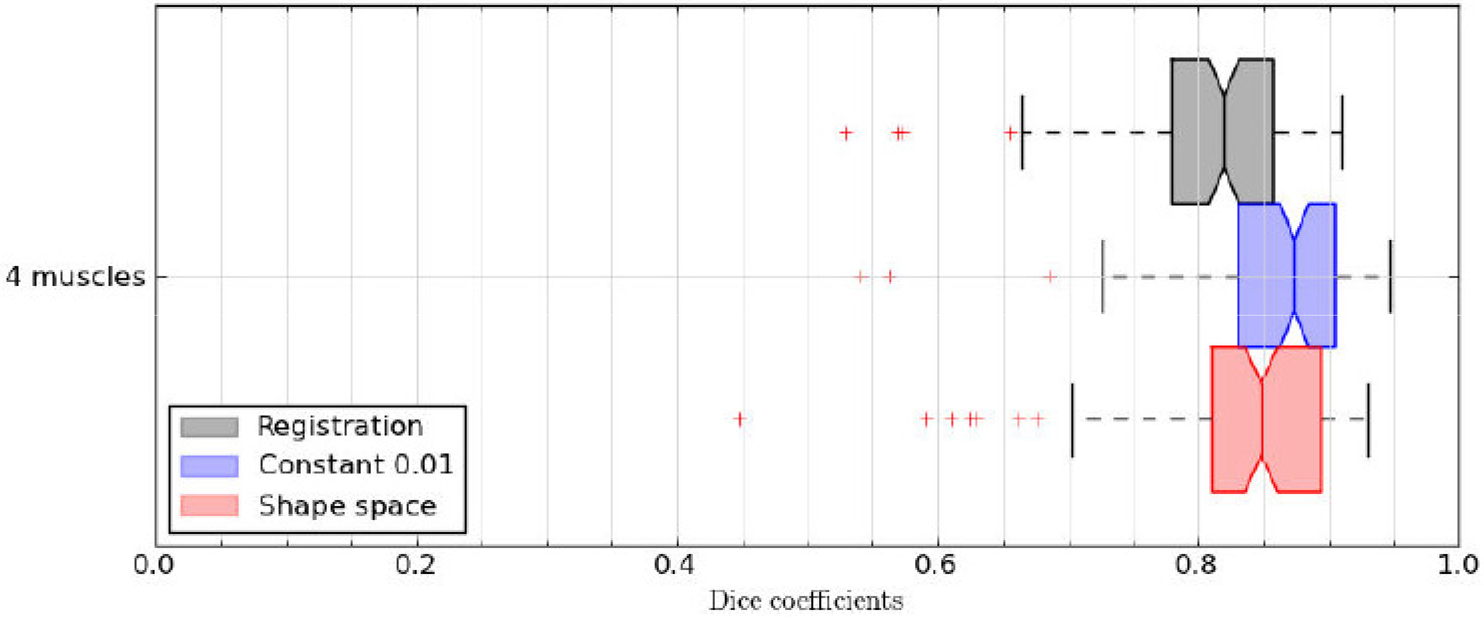
where I_{KN} is the identity matrix of size $KN \times KN$.

The minimum of (5.4.16) verifies:

$$\left(A^T\widetilde{L}A + \lambda_{\mathbf{dx}}B\right)\mathbf{y} = -A^T\widetilde{L}\mathbf{x}_0. \qquad (5.4.18)$$

RW & Prior Shape Model (Baudin et al, BMVC 2012)

Experimental results



RW & Prior Shape Model (Baudin et al, BMVC 2012)