Proceedings of the 28th IEEE
EMBS Annual International Conference
New York City, USA, Aug 30-Sept 3, 2006

A Visual Validation Method Based on Point Mappings for Medical Image Segmentation

Feng Qian, Lixu Gu *Member, IEEE*, Shaoting Zhang

Abstract—**Validation for image segmentation plays an important role in medical image processing. Point-mappings based validations process segmentation results by matching corresponding points on surfaces of the segmentation model and the standard model, then evaluate the matches locally and globally. This paper proposes four algorithms of building point-mappings; and creatively applies perfect matching algorithm of weighed bipartite graph on it. Furthermore, detailed local and global evaluation methods are also presented. Finally, experiments demonstrate that such validation method is competent to evaluate 2D or 3D segmentation scientifically, expeditiously and accurately.**

I. Introduction

Segmentation, which picks up regions of interests (ROI) from the source images, is a key technology in medical image processing field. Large amount of work have been carried out and thousands of segmentation methods have been obtained [1]. Its validation, however, does not receive its deserved recognition; the only tens of validation methods can be grouped into four kinds:

(1) No validation, or only visual inspections by people [2];
(2) ***Qualitative analysis*** (such as apriori information [3] and strategies) or quantitative analysis (such as complexity and cost) of segmentation methods; they lack experiments of actual segmentation results;
(3) ***Global quantitative experiments*** (such as comparing sizes, volumes, overlapping areas and proportion of correct pixels [4], between standard and segmentation models); they merely use several global functions as indicators and detailed local information are lost;
(4) ***Local quantitative experiments*** (such as method based on radial distance errors [4]); some of them require generating sample points manually; and often the results are not accuracy enough.

The method proposed here is a full automatically visual validation based on (both local and global) quantitative experiments [5]. The inputs are two digital organic models, G and S, which are composed of mass of points on their surfaces. G is the "Gold Standard" [6], which is the standard model segmented by experienced clinical doctors; S is the machine-segmented model to be validated. Our task is to find the difference between G and S, which measures the performance of segmentation algorithms. [2]

We make two reasonable assumptions: only points on surfaces are taken into consideration; numbers of points on surfaces of G and S are both n[1]. Then, the whole validation can be divided into three steps:

(1) ***Building mappings***: Build point-mappings from G to S, every point on G are mapped to the "most matched" point on S.
(2) ***Local Evaluation***: Use a local evaluation function to grade each mapping built in (1).
(3) ***Global Evaluation***: Use global evaluation functions to grade the matching degree of the whole model by synthesizing all mappings (and their local evaluations).

In this paper, we will propose four algorithms of building point mappings (§2), we also suggest methods of local and global evaluation (§3); finally experiments and results are presented (§4).

II. Algorithms Of Building Point-mappings

A. Common Objectives

Building point-mappings is to create matches between corresponding points on surfaces of G and S[2]; it's the key process of the validation. Its common objectives are (Fig. 1):

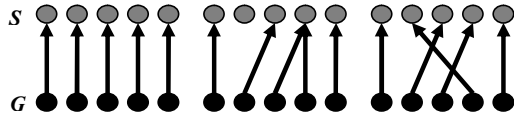

Fig. 1. Well-ordered mappings, Null mappings and Cross mappings

(1) Each point on G is mapped to at least one point on S;
(2) Try to keep homogeneous: As many as possible points on S should be mapped, and prevent "null mappings" ;
(3) Try to keep consecutive: The directions[3] of mappings

Manuscript received April 3, 2006. This work is partially supported by the Shanghai municipal research fund (045115002 and 045107045) and the Natural Science Foundation of China (Grant No. 70581171).

All authors are with Lab of Image Guided Surgery and Therapy, Shanghai Jiaotong University, 800 Dongchuan Rd, Shanghai, China., 200240
First Author Feng Qian, E-mail: shinyflute@gmail.com
Second Author Lixu Gu, E-mail: gu-lx@cs.sjtu.edu.cn
Third Author Shaoting Zhang, E-mail: kawaii.tony@gmail.com

[1] This is only for simpleness of representation; there's no such limit in actual application

[2] Uppercases G and S denote standard and segmentation model, respectively; lowercases g and s denote points on them

should be as consist as possible in a local spot, and prevent "cross mappings".

B. Two Simple methods: Closest-Point Algorithm and Ordinal-Inserting Algorithm

Closest-Point Algorithm (CPA) employs a point with minimal Euclidean distance on *S* as the mapped point for each point on *G*[4]. It's simple and fast; however, the distinct drawback is the non-homogenization in those "scraggy" areas on *S* [7]. In fact, many known methods [8] merely use this approach; then distortions are inevitable (Fig. 2).

```
for i = 1 to n do
  Find s_i^* ∈ S, that dist(g_i, s_i^*) = min_{s∈S}{dist(g_i, s)}
  Add {g_i → s_i^*} to M
end for
```

Algorithm 1. Closest-Point Algorithm

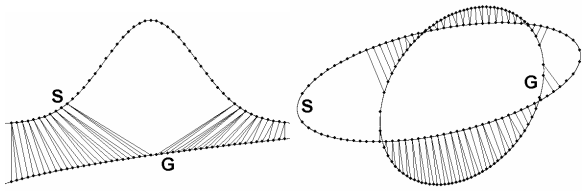

Fig. 2. Simulation of Closest-Point Algorithm

Ordinal-Inserting Algorithm (OIA) sorts possible n^2 mappings ascendantly by distance, then adds them into mapping set *M* one by one. To keep homogeneous, one inserting mapping is skipped if its either point (*g* or *s*) is already mapped. This algorithm is also straightforward and high-efficient; however, it drawback is: mappings of long distances are inserted at a late stage, when most points are already occupied by shorter mappings. Such few choices lead to considerable error for long mappings (Fig. 3); so corrections are necessary (§2.E).

```
mapList = empty list
for i = 1 to n do
  for j = 1 to n do
    Insert {g_i, s_j, dist(g_i, s_j)} into mapList
  end for
end for
Sort mapList by distance ascendently
for t = 1 to n^2 do
  Get next tuple of {g, s, dist(g, s)} from mapList
  if (not mapped(g)) and (not mapped(s)) then
    Add {g → s} to M
end for
```

Algorithm 2. Ordinal-Inserting Algorithm

C. Reversed-Inserting Algorithm

Reversed-Inserting Algorithm (RIA) is a revised version of Ordinal-Inserting Algorithm. It endows long mappings with high priorities. First, sort *2n* points in *G* and *S* by their minimal distance to the other model (*S* and *G*, respectively) descendently. Second, process these points (ordered by minimal-distance from long to short): for each unmapped point *p* belongs to *G* or *S*, use its closest unmapped point on *S* or *G*, respectively, as the mapped point; note that for each *p*, the length of inserted mapping should also be descendent, if all mappings checked are longer than the previous inserted mapping, then *p* is skipped. Finally, there may be a small number of unmapped points on *G*, then closest-point algorithm may directly apply to them. In Fig. 4, we see that its performance is much better than CPA and OIA. Experiments show that its accuracy sufficiently meets requirements of most validations. (§ 4)

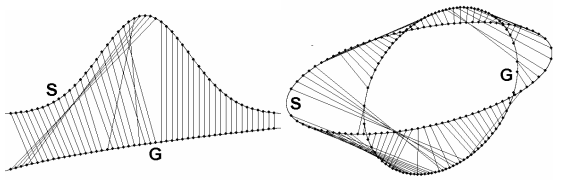

Fig. 3. Simulation of Ordinal-Inserting Algorithm

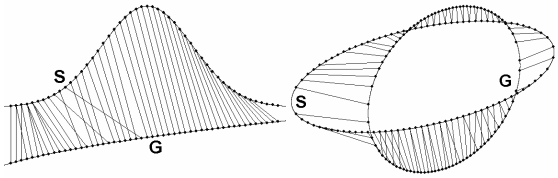

Fig. 4 Simulation of Reversed-Inserting Algorithm

D. Perfect Matching on Weighed Bipartite Graph

By applying ***Perfect Matching Algorithm of Weighted Bipartite Graph*** (PMB) on point-mapping problem, we find that the result is almost ideal[5]. The novel method focuses on the sum of all mappings instead of a single one. Consider the following function:

$$C(f) = \sum_{i=1}^{n} dist(g_i, s_{f(i)}) \quad (1)$$

The independent value *f* is a bijection from *{1..n}* to a permutation of *n*. When *C(f)* reaches its minimum value, the sum of distances of bijections from *G* to *S* is also minimal; at this state, mappings (described by *f*) tends to be ideal matches (Fig. 5). In fact, finding minimum value of (1) and solving the perfect matching problem of weighed bipartite

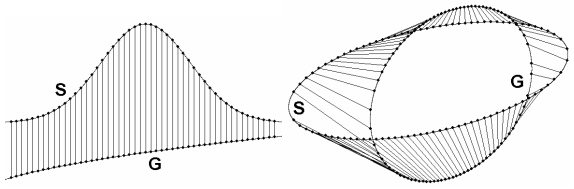

Fig. 5. Simulation of Perfect Matching Algorithm of Weighed Bipartite Graph

[3] The direction of a mapping from *g* to *s* is defined as the direction of the radial from *g* to *s*

[4] In the pseudo code, *M* denotes the mapping set; $g \rightarrow s$ denotes a mapping from *g* to *s*; function *dist* is the Euclid distance between two points

[5] Another known application of perfect matching on weighed bipartite graph is shot retrieval [9]

ptrList = empty list
for $i = 1$ *to* n *do*
  Find s_i^* *that* $dist(g_i, s_i^*) = \min_{s \in S}\{dist(g_i, s)\}$
  Find g_i^* *that* $dist(g_i^*, s_i) = \min_{g \in G}\{dist(g, s_i)\}$
  Insert $\{g_i, dist(g_i, s_i^*)\}$ *and* $\{s_i, dist(g_i^*, s_i)\}$ *into ptrList*
end for
Sort ptrList by distance descendently, $preDist = +\infty$
for $t = 1$ *to* $2n$ *do*
  Get next tuple of $\{p, dist(g,s)\}$ *from ptrList*
  if $(not\ mapped(p))$ *and* $(p \in G)$ *then*
    $\Psi_S = \{s \mid not\ mapped(s)\ and\ s \in S\ and\ dist(p,s) < preDist\}$
    if $(\Psi_S \neq \varnothing)$ *then*
      Add $\{p \to s^*\}$ *to* M, *where* $s^* \in \Psi_s$ *and* $dist(p, s^*)$ *is minimal*
      $preDist = dist(p, s^*)$
    end if
  else if $(not\ mapped(p))$ *and* $(p \in S)$ *then*
    $\Psi_G = \{g \mid not\ mapped(g)\ and\ g \in G\ and\ dist(g,p) < preDist\}$
    if $(\Psi_G \neq \varnothing)$ *then*
      Add $\{g^* \to p\}$ *to* M, *where* $g^* \in \Psi_G$ *and* $dist(g^*, p)$ *is minimal*
      $preDist = dist(g^*, p)$
    end if
  end if
end for
for $u = 1$ *to* n *do*
  if $(not\ mapped(g_u))$ *then*
    Find s_u^* *that* $dist(g_u, s_u^*) = \min_{s \in S}\{dist(g_u, s)\}$, *Add* $\{g_u \to s_u^*\}$ *to* M
  end if
end for

Algorithm 3. Reversed-Inserting Algorithm

graph are isomorphic. A classical algorithm solving it is proposed by *Kuhn* [11]. Many much better algorithms are available. [10] is a good summary of them.

E. Improvements and correction of algorithms

In a mapping $g \to s$, the distance between g and s may not be very far, so it's reasonable to just concern the nearest k points (on S) from g; thus, the complexity of enumeration is decreased from n^2 to $n \cdot k$ ($k \ll n$).

In order to rectify inconsecutive mappings in Algorithm OIA and RIA, when we encountered following mappings, they may be skipped:

$$\{g \to s \mid \frac{\overrightarrow{N_g} \cdot \overrightarrow{(g,s)}}{dist(g,s)} < \delta\}$$

The two vectors are the normalized vector of the surface of g and the vector from g to s. If their included angle is larger than $arccos(\delta)$ then the mapping tends to be not reasonable. An alternative way is to consider the difference between the direction of an inserting mapping $g \to s$, and the average direction of inserted mappings around g.

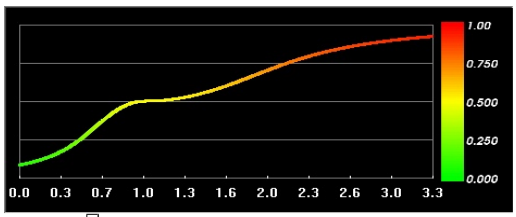

Fig. 6. Use 3 as $\varphi(x)$ ($x_{perfect}$=0.1, x_{accept}=1.0, $x_{unaccept}$=3.0, ζ=0.1)

III. Local and Global Evaluations

A. Local Evaluation Function

Local evaluation function E grades a single mapping (built by a point-mapping algorithm) by a real number between 0 and 1:

$$E : (g \to s) \to \{y \mid 0 < y < 1, y \in \Re\}$$

$E(g \to s)$ composes of two parts: (1) I_s: The self-attribute of g and s; (2) I_r: Relationships between g and s. For (1), we can assign a weight, indicating the degree of our attention, to each point on a model (usually G). Thus we can increase or decrease our required segmentation accuracy of particular points; For (2), usually, we use the distance between g and s as the basis for evaluations (other attributes may also be used, like the included angle mentioned in §2.E). Finally, we map the product of I_s and I_r to *(0, 1)* by a mapping function φ to get the value of E: (φ:$\boldsymbol{R} \to (0,1)$)

$$E(g \to s) = \varphi(I_s(g,s) I_r(g,s)) = \varphi(I_s(g,s) dist(g,s))$$

B. The Mapping Function φ

$\varphi(x)$ maps any real number to *(0, 1)*. We assign three threshold values for x: $x_{perfect}$, x_{accept} and $x_{unaccept}$.[6]

$$\begin{cases} \varphi(x_{perfect}) = \xi \\ \varphi(x_{accept}) = 1/2 \\ \varphi(x_{unaccept}) = 1 - \xi \end{cases} \quad ,\xi \in (0, 1/2) \qquad 2$$

ζ is ***accuracy threshold***, for qualitative analysis:

$$\begin{cases} 0 < \varphi(x) \le \xi : \textit{ The mapping is perfect} \\ \xi < \varphi(x) \le 1 - \xi : \textit{ The mapping is acceptable} \\ 1 - \xi < \varphi(x) \le 1 : \textit{ The mapping is not acceptable} \end{cases}$$

Thus, $\varphi(x)$ should satisfy Equation 2.

3 is a sample $\varphi(x)$ found by us (Fig. 6). One of its important characteristics is $\partial\varphi / \partial x(x_{accept}) \equiv 0$.

$$k_1 = \tan\left(\pi\left(\xi - \frac{1}{2}\right) \Big/ \left(x_{accept} - x_{perfect}\right)^2\right)$$
$$k_2 = \tan\left(\pi\left(\frac{1}{2} - \xi\right) \Big/ \left(x_{unaccept} - x_{accept}\right)^2\right) \qquad 3$$
$$\varphi(x) = \begin{cases} \arctan(k_1 (x - x_{accept})^2) / \pi + 1/2 & (0 < x \le x_{accept}) \\ \arctan(k_2 (x - x_{accept})^2) / \pi + 1/2 & (x_{accept} < x) \end{cases}$$

C. Under-Segmentation and Over-Segmentation

Under-Segmentation indicates one point s on S is outside the outline of G; ***Over-Segmentation*** suggests s is inside G. We modify E to E^* to describe them:

$$E^* = \begin{cases} E(Over\ Segmentation) \\ 0(Accurate\ Segmentat \\ -E(Under\ Segmentati \end{cases}$$

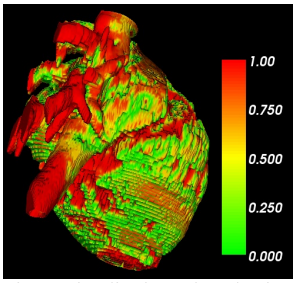

Fig. 7. Visualized Local Evaluation

[6] [12] and [4] inspire me

D. The Visualization of Local Evaluation

We map local evaluation results (E, E^*) to a color table (Fig. 6), then display G with its points colored. Thus, results are clear at a glance[7] (Fig. 7). Another choice is to calculate numbers or percentages of points within different accuracy intervals, then plot histograms [7] or sector diagrams.

E. Global Evaluation

Global Evaluation grades all mappings quantitatively. Some frequently used global evaluation functions are:

(1) Homogenization of Segmentation Model (***HSM***);

$$HSM = \sqrt{\sum_{i=1}^{n}\left(MinDist(S_i) - \frac{1}{n}\sum_{j=1}^{n} MinDist(S_j)\right)^2} \Big/ \sum_{i=1}^{n} MinDist(s_i)$$

(2) Average/Maximum/Minimum/Standard Deviation of Local Evaluation (***LE***) Value or Error Distance (***ED***);

(3) Percentage of Perfect/Acceptable/Unacceptable Mappings (***PPM***, ***PAM***, ***PUM***);

(4) Percentage of Over-Segmentation/Under-Segmentation (***POS***, ***PUS***);

IV. Experiments and Results

We use data of three organs (brain, heart and kidney) generated from CT. (Table 1). For each set of data, we use four point-mapping algorithms introduced in §2, respectively, and the same local (global) evaluation method (§3) to validate the segmentation. The experiment results of the third data set (kidney) are excerpted in Table 2.[8]

In experiments, CPA ignores the consistency of point-mappings, thus exaggerates the accuracy considerably;

Organ	Segmentation Algorithm of S	Segmentation Method of G	$n_s : n_g$
Brain	Fast Marching	Segmented by experienced doctors	8125:8563
Heart	Morphological Reconstruction		6510:6439
Kidney	Watershed		8360:8826

Table 1. Experiment source data

redundant green areas (accurate-segmented areas) can be seen in Fig. 8-1. Noises in last stages of OIA leads to the excessive PUM rate and blue areas (imprecisely-segmented areas) in Fig. 8-2; errors between RIA and PMB are acceptable except the time cost, which limits the application of PMB. Results of other two organs are similar with the kidney case. As the conclusion, we summary four algorithms in Table 3.[9]

Alg.	Performance	Speed	Complexity	Applications
CPA	Basic and rough	Very Fast	$O(n)$	Real-time validation
OIA	Worse than RIA	Fast	$O(knlogkn)$	General applications
RIA	Practical	Fast	$O(nlogn+kn)$	
PMB	High accurate and complicated	Very Slow	Kuhn [11]: $O(kn^3)$ Best: $O(kn^{3/2}lognU)$	small models; "Gold Standard" of point mapping algorithms;

Table 3. Summary of four point mapping algorithms

Algorithm	CPA	OIA	RIA	PMB
Time	1sec	19sec	24sec	13hour
AVGLE	0.4435	0.5018	0.4986	0.4974
AVGED	0.4683	0.5875	0.5766	0.5751
PPM	15.89%	12.64%	11.43%	11.57%
PUM	5.81%	10.01%	8.66%	8.54%
POS	90.13%	89.87%	90.25%	90.36%
PUS	9.87%	10.13%	9.75%	9.64%

Table 2. Experimental results of the kidney model (Fig. 8-1 to Fig. 8-4) CPA, OIA, RIA and PMB are four algorithms mentioned in §2

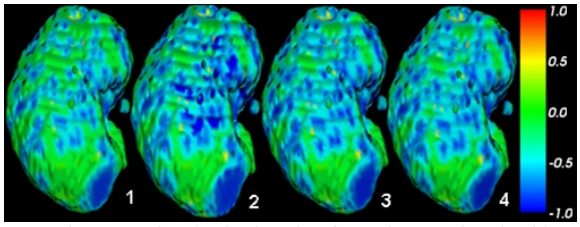

Fig. 8. Local evaluation based on four point-mapping algorithms

[7] [8] and [4] inspire me

[8] In Table 2, please refer to §3.E for definitions of *AVGLE*, *AVGED*, *PPM*, *PUM*, *POS* and *PUS*

[9] In Table 3, please refer to §2.E for the definition of k

References

[1] *Dzung L. Pham, Chenyang Xu, Jerry L. Prince*, "A survey of current methods in medical image segmentation", *Vol.2:315-337*, *Annual Review of Biomedical Engineering, 1998*

[2] *C. N. de Graaf, A. S. E. Koster, K. L. Vincken, M. A. Viergever*, "A Methodology For the Validation of Image Segmentation Methods", *Computer-Based Medical Systems, 1992. Proc., Fifth Annual IEEE, 17-24*

[3] *C. E. Liedtke et al.*, "Segmentation of Microscopic Cell Scenes", *AQCH, 1987,9:197-211*

[4] *JianFeng Xu, Lixu Gu*, "A novel validation method based on radial distance error for 3D medical image segmentation" *Journal of Biomedical Engineering (in press)*

[5] *Yujin Zhang*, "A classification and comparison of evaluation techniques for image segmentation" *China Journal of Image and Graphics, Vol. 1, No. 2 June, 1996*

[6] *Paulo Correia, Fernando Pereira*, "Objective evaluation of relative segmentation quality" *Proc. of IEEE Conference on Image Processing (ICIP2000) Canada, 10-13, Sept. 2000: 308-311*

[7] *Pascal Bertram, LiXu Gu, JianFeng Xu*, "Segmentation accuracy calculation based on local and global errors" *CARS-2006, Osaka.(in press)*

[8] *Araon Fenster, Bernard Chiu*, "Evaluation of Segmentation algorithms for Medical Imaging" *Proc. of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*

[9] *Yuxin Peng, NGO Chong-Wah, Jianguo Xiao*, "An approach for shot retrieval by optimal matching in the bipartite graph" *Acta Electronica Sinica, Vol.32 No.7 July 2004*

[10] *Herbert Alexander Baier Saip, Claudio Leonárdo Lucchesi*, "Matching Algorithms for Bipartite Graphs" *Relatório Técnico DDC-03 93*

[11] *H. W. Kuhn*, "The Hungarian method for the assignment problem", *Naval Res. Logist. Quart., 2:83-97, 1955*

[12] *Lixu Gu, Jianfeng Xu, Terence M. Peters*, "A novel multistage 3D medical image segmentation: methodology and validation" *Computational Intelligence and Security. Lecture Notes in Computer Science(LNCS), Vol.3801 (2005)884-889*