

به نام خدا



دانشگاه صنعتی شریف

درس پردازش و تحلیل تصاویر پزشکی

پروژه شبیه سازی مقاله:

" Image registration by local histogram matching"

By: Dingang Shen

عبداله قنبری

۹۰۲۰۶۸۶۹

استاد راهنما: دکتر فاطمی زاده

تابستان ۹۱

شرح مسئله و چالش‌های آن

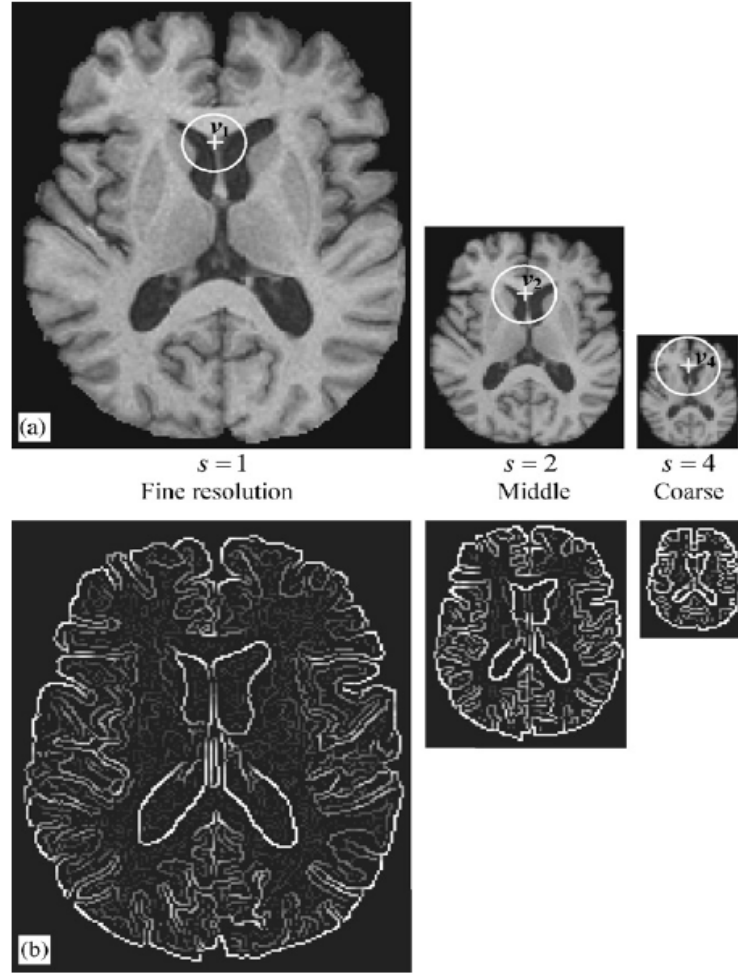
انطباق تصاویر خصوصاً در تصاویر پزشکی به ویژه در تصاویر MRI از پیچیدگی و حساسیت بالایی برخوردار است. پیچیدگی موجود در این تصاویر باعث می‌شود تا از الگوریتم‌هایی با دقت بالا برای انطباق بهره گرفته شود، هرچند مشکلی که این الگوریتم‌های دقیق به وجود می‌آورند انطباق تصاویر با سرعت‌های بسیار پایین می‌باشد که عملاً پیاده‌سازی این الگوریتم‌ها را با دشواری روبرو می‌سازد. در مقاله‌ی حاضر به ازای هر نقطه از تصویر بردار صفاتی تشکیل می‌شود که برای سرعت دهی به فرایند محاسبه‌ی این بردار صفات از هیستوگرام به عنوان سهل‌الوصول‌ترین و سریع‌ترین فضای محاسبه برای استخراج این صفات استفاده شده است. ویژگی دیگر مورد نظر در اینجا که ما را به هیستوگرام علاقه‌مند می‌کند عدم وابستگی به جهت‌دهی تصویر می‌باشد که جابجایی پیکسل‌ها را در هر جهت برای ما میسر می‌کند، به بیان دیگر می‌توان گفت که بردار صفات ساخته شده از چرخش تصویر مستقل است. ویژگی دیگر این روش، توجه همزمان آن به ویژگی‌های محلی، میانه و کلی تصویر می‌باشد. بنابراین می‌توان گفت عمده هدف این مقاله سرعت دهی به فرآیند محاسبه ویژگی‌ها و استفاده از ویژگی‌های مکانی و مقادیر سطوح خاکستری می‌باشد. از سوی دیگر مقاله‌ی آقای Shen ادامه کار قبلی ایشان در الگوریتم HAMMER می‌باشد به گونه‌ای که سعی کرده مشکل اساسی HAMMER یعنی Pre-Segmentation را در این روش مرتفع کند تا بتواند با حذف این مرحله از این الگوریتم برای تصاویر غیر-مغزی هم استفاده کند زیرا مرحله Segmentation برای هر ارگان و هر مدالتی تصویربرداری متفاوت خواهد بود. به طور کلی هدف اساسی این مقاله را می‌توان در استفاده از اطلاعات هیستوگرام به تنهایی برای انطباق تصاویر بر هم بیان کرد.

شرح الگوریتم

الگوریتم را می‌توان در دو مرحله خلاصه کرد به گونه ای که در مرحله اول شباهت میان دو تصویر سنجیده می‌شود و در مرحله دوم تغییر شکل تصویر مبدأ با معیار های تعریف شده بر اساس شباهت به دست آمده در مرحله اول صورت می‌گیرد.

الف (محاسبه بردار صفات : بررسی شباهت میان دو تصویر از روی مقایسه ویژگی‌های بدست آمده از هیستوگرام می‌باشد. ویژگی‌های بدست آمده را می‌توان به دو دسته ویژگی تقسیم کرد دسته‌ی اول ویژگی‌های تصویر اصلی و دسته‌ی دوم ویژگی‌های حاصل از تصویری مشتقه از تصویر اصلی می‌باشد. تصویر دوم می‌تواند شامل لبه های تصویر، گرادیان و یا تعریف‌هایی دیگر از تصویر مبدأ باشد. مرحله‌ی اول یا همان بررسی شباهت دو تصویر در چند گام صورت می‌پذیرد، همان‌طور که در شکل ۱ نیز نشان داده شده است در گام اول سه مقیاس از تصویر اصلی که هر یک با نسبت $s=1,2,4$ Down-sample شده‌اند به وجود می‌آوریم، همین عمل را برای تصویر لبه‌ها نیز انجام خواهیم داد. در هر یک از این تصاویر و برای هر یک از پیکسل‌های این تصاویر دایره ای به شعاع R حول پیکسل مورد نظر خواهیم داشت که هیستوگرام مورد استفاده برای هر پیکسل فقط از ناحیه داخل دایره استفاده می‌کند. توجه داریم که شعاع دایره برای تمام تصاویر از جمله تصویر با مقیاس $s=4$ نیز ثابت است به این ترتیب هیستوگرام در هر مقیاس اطلاعات مشخصی از اطراف تصویر را به وسیله هیستوگرام حول آن نقطه به همراه خواهد داشت. به عبارت دیگر در این سه مقیاس اطلاعات جزئی، میانه و کلی با هم دیده می‌شود. در گام بعد از این هیستوگرام‌ها برای هر یک از ۶ تصویر حاصل ۳ ممان خواهیم داشت که شباهت هیستوگرام‌ها از روی مقایسه این ممان‌ها صورت می‌پذیرد. هر یک از ممان‌ها به صورت رابطه (۱) از روی هیستوگرام تصاویر بدست خواهد آمد.

$$(1) \quad m(v_s, p) = \sum_i i^p h_s(v_s, i)$$



شکل (۱): نحوه‌ی محاسبه هیستوگرام در رزولوشن‌های مختلف و برای دو تصویر اصلی و تصویر لبه‌ها

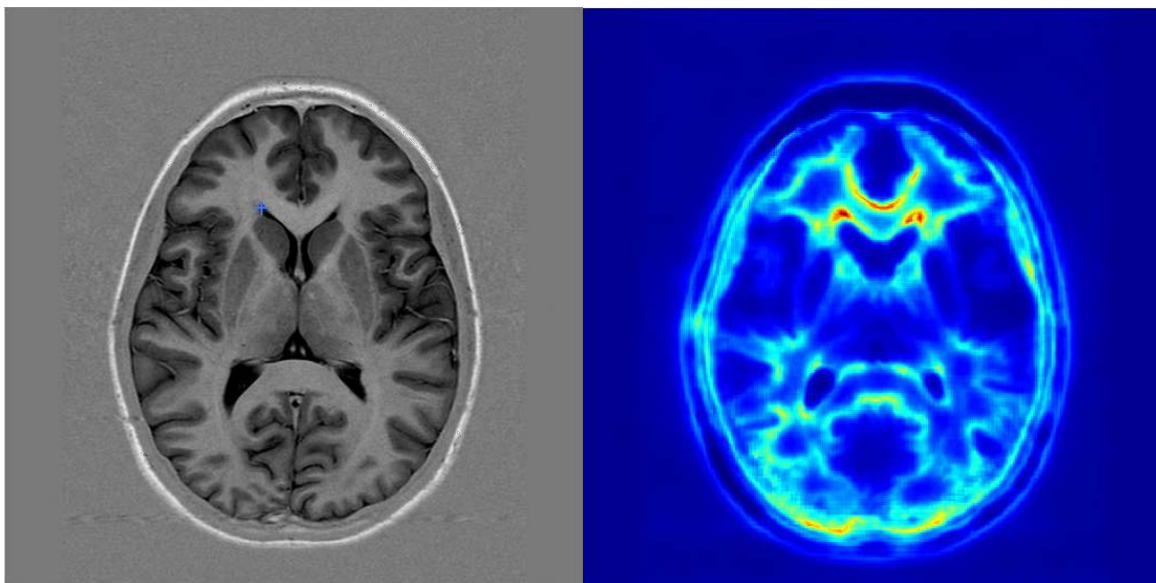
که در رابطه‌ی تعریف شده $h_s(v_s, i)$ فراوانی بین i ام و p ممان هیستوگرام می‌باشد. با تعریف ممان به این صورت برای هر پیکسل در هر یک از این ۶ تصویر سه ممان در نظر می‌گیریم و در نهایت ماتریسی از بردارهای ممان تصویر اصلی و تصویر مرزها در ۳ مقیاس خواهیم داشت که در ماتریس $a(v)$ که نمایشگر ماتریس صفات پیکسل می‌باشد قرار می‌دهیم.

$$(2) \quad a(v) = [[a_1^{Hist}(v) b_1^{Bound}(v)], [a_2^{Hist}(v) b_2^{Bound}(v)], [a_4^{Hist}(v) b_4^{Bound}(v)]]$$

از بردار صفات حاصله از پیکسل v برای مقایسه با پیکسل u از رابطه‌ی زیر بهره می‌گیریم که در واقع شباهت این دو را می‌سنجد.

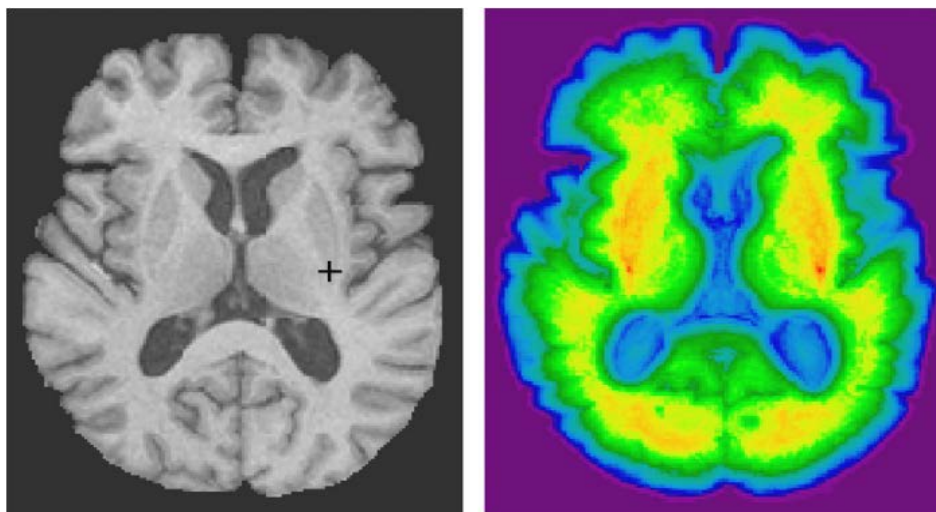
$$(3) \quad m(a(v), a(u)) = \prod_s (1 - |b_s^{Bound}(u) - b_s^{Bound}(v)|) * \prod_i (1 - |a_{s,i}^{Hist}(u) - a_{s,i}^{Hist}(v)|)$$

با تعریف صورت گرفته می‌توان نشان داد که هیستوگرام های بدست آمده نسبت به چرخش تصویر مستقل می‌باشند همچنین می‌توان با نرمالیزه کردن هیستوگرام ها ناهمگنی روشنایی در دو تصویر را برطرف کرد. تمایز پذیری نقاط توسط مقایسه ماتریس صفات دو پیکسل از تصویر صورت می‌پذیرد به عنوان مثال هر پیکسل از تصویر مبدأ (شکل ۲) با تمامی نقاط دیگر همان تصویر مقایسه شده است و تصویر حاصل به صورت یک تصویر رنگی نمایش داده شده است به گونه ای که نقاط قرمز رنگ نمایش دهنده بیشترین شباهت و نقاط آبی کمترین شباهت را نشان می‌دهند.



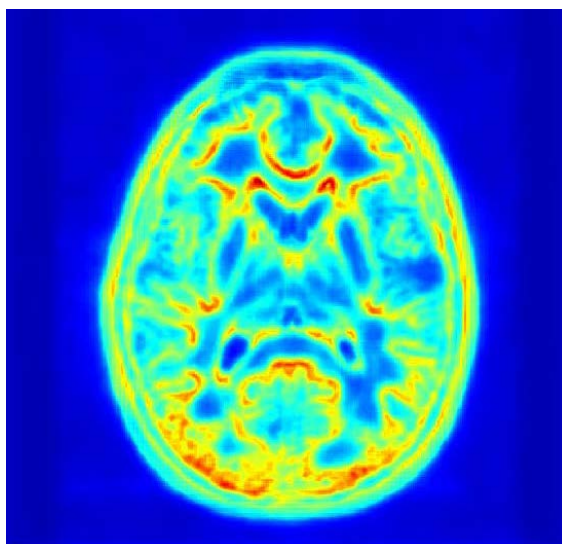
شکل ۲: تصویر اولیه و نقطه‌ای مورد مقایسه (سمت چپ) و نمایش شباهت آن با بقیه نقاط تصویر به صورت یک تصویر کد شده رنگی (سمت

راست)



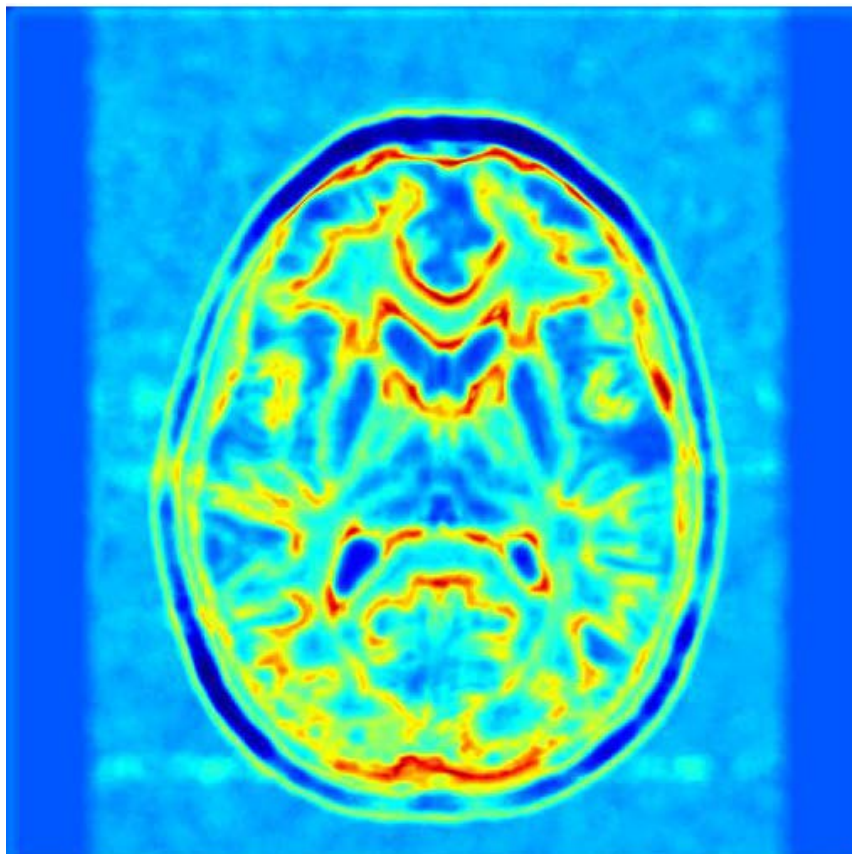
شکل ۳: تصویر اولیه مورد استفاده در مقاله مرجع و نقطه‌ی مورد مقایسه (سمت چپ) و نمایش شباهت آن با بقیه نقاط (سمت راست)

همان طور که در تصاویر نیز مشخص است تصاویر حاصل در پیاده سازی الگوریتم نسبت به مقاله نتایج بهتری را نشان می‌دهد که به علت به کارگیری ممان های بیشتر می‌باشد همچنین نحوه‌ی رنگ بندی تصویر در مرجع با تصویر نمایش داده شده در شکل (۲) متفاوت است. در صورتی که از ممان های کمتری استفاده کنیم نقاط مشابه تصویر بسیار بیشتر خواهد شد به عنوان مثال با در نظر گرفتن تنها ۲ ممان تصویر حاصل از میزان شباهت در شکل (۴) نمایش داده شده است.



شکل ۴: شباهت پیکسل منتخب با پیکسل‌های دیگر تصویر با استفاده از ۲ ممان

در شکل (۵) تنها از تصویر $S=2$ یعنی همان تصویر میانه استفاده شده که همان گونه که نشان داده شده است توانایی تمایز پذیری به خوبی استفاده از هر سه تصویر را ندارد.



شکل ۵: تمایز پذیری تنها با استفاده از اطلاعات تصویر میانه

تا به اینجا کارایی الگوریتم و قابلیت و نیاز مراحل تعریف شده برای الگوریتم توضیح داده شده است در مرحله بعد توضیح مختصری در مورد تابع هدف تعریف شده بیان می کنیم. تابع هدف تعریف شده به صورت رابطه ی زیر می باشد.

$$E = \sum_u \omega_T(u) \left(\frac{\sum_{z \in n(u)} \varepsilon(z) (1 - m(a_T(z), a_S(z)))}{\sum_{z \in n(u)} \varepsilon(z)} \right) + \sum_v \omega_S(u) \left(\frac{\sum_{z \in n(v)} \varepsilon(z) (1 - m(a_T(h^{-1}(z)), a_S(h^{-1}(z))))}{\sum_{z \in n(v)} \varepsilon(z)} \right) + \beta \sum_u ||\nabla^2 d(u)||$$

تابع هدف تعریف شده قصد دارد تا ماتریس صفات نقاط متناظر در تصویر را بر هم منطبق کند. ترم اول این تابع هدف مجموع وزن دار شباهت هر پیکسل با نقاط همسایه آن در تصویر مقصد می باشد. ترم دوم نیز همین عمل را به صورت عکس انجام می دهد یعنی مجموع وزن دار شباهت نقاط همسایگی یک پیکسل در تصویر مبدأ با نقطه متناظر در تصویر مقصد می سنجد. هر دوی این ترم ها نرمالیزه شده اند. ترم سوم نیز همواری میدان جابجایی را چک می کند.

(ب) تبدیل غیر-صلب B-Spline: در اینجا از یک روش Point-Based Registration استفاده می کنیم به گونه ای که مسئله به صورت یافتن تبدیلی طرح می شود که بتواند در فضای دو بعدی مجموعه ی نقاط متناظر در دو تصویر را به یکدیگر بنگارد.

ورودی: n جفت از نقاط مبدأ و مقصد $(p_i, q_i) \in \mathbb{R}^2, i = 1, \dots, n$.

خروجی: یک تابع دو بعدی $f: \mathbb{R}^2 \rightarrow \mathbb{R}^2, f(p_i) = q_i, i = 1, \dots, n$

در واقع مسئله به دو دسته معادلات درون یابی داده های پراکنده که دسته ی (p_i, q_i^x) در راستای x و (p_i, q_i^y) در راستای y می باشند تبدیل می شود برای حل این مسئله توابع درون یابی مکانی متفاوت پیشنهاد شده است که تبدیل غیر صلب B-Spline یکی از این راه حل ها می باشد.

برای بدست آوردن تبدیل مورد نظر با داشتن نقاط متناظر از بسته کدهای آماده موجود در [۲] استفاده شده است.

تجربیات ناموفق/موفق شبیه سازی

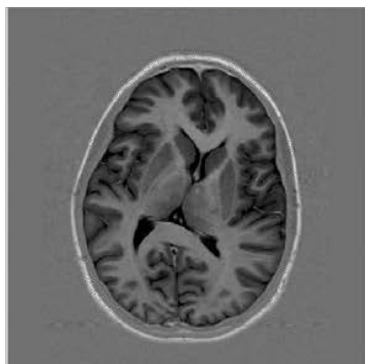
نکته‌ی قابل ذکر در اینجا این است که روند کار در ادامه پروژه‌ی حاضر با روند کار در مقاله آقای Shen متفاوت است به گونه‌ای که در مقاله‌ی آقای Shen به بیان اهداف اصلی طرح تابع هدف و معیارهای انتخاب این تابع هدف که در واقع همان تابع هدف بکار رفته در الگوریتم Hammer می‌باشد اکتفا کرده است و صحبتی از نحوه‌ی پیاده‌سازی آن به میان نیاورده بود بنابراین به نظر می‌رسید تنها راه پیاده‌سازی الگوریتم محاسبه مقدار انرژی برای هر یک از نقاط به صورت جداگانه بود و همان‌طور که انتظار می‌رفت پیاده‌سازی چنین الگوریتمی در متلب به فضای حافظه و زمان بسیار زیادی نیاز داشت زیرا به ازای هر یک از نقاط در تصویر به فضای ماتریسی همانند آنچه در شکل ۲ نمایش داده شده بود نیاز داشتیم. بنابراین در نحوه‌ی پیاده‌سازی تغییراتی ایجاد کردیم به گونه‌ای که اگر مقاله را به دو بخش یافتن نقاط متناظر و اعمال تابع هدف تقسیم کنیم در بخش اعمال تابع هدف تغییراتی ایجاد کردیم بدین صورت که نقاطی از تصویر مبدأ را در نظر گرفتیم و از این نقاط به عنوان نقاط اولیه تصویر نام بردیم و در یک همسایگی از این نقطه، در تصویر مقصد به دنبال نقطه‌ای که بیشترین شباهت را با نقطه‌ی اولیه در تصویر مبدأ داشته باشد می‌گردیم اگر نقطه مورد نظر در همسایگی‌ای نزدیک‌تر از همسایگی تعریف شده اولیه قرار داشت نقاط مبدأ و مقصد را به عنوان نقاط متناظر با یکدیگر نگه خواهیم داشت و از این نقاط برای اعمال تبدیل B-Spline استفاده می‌کنیم این عمل را تا به آنجایی ادامه خواهیم داد که در نهایت خطای میان تصویر اولیه و تصویر ثانویه از یک حد از قبل مشخص شده‌ای کمتر شود. در شکل‌های زیر تصویر اولیه و تصاویر ثانویه در هر گام به نمایش درآمده است.

الگوریتم زیر مراحل کار را نمایش داده است که نحوه‌ی کارکرد روش پیشنهادی را از ابتدا تا پایان به نمایش در آورده است.

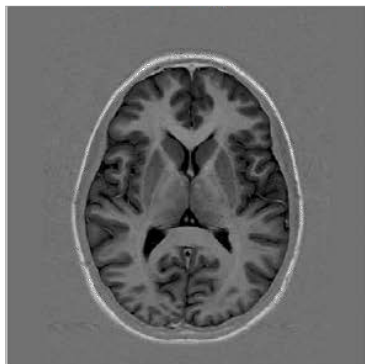
Algorithm: Automatic Histogram-based Landmarks Extraction and Reduction coped with B-spline Registration	
1 :	read two images;
2 :	static image multi-resolution and edge creation;
3 :	create features for each points of images in 2;
4 :	while (<i>difference of two images</i> < <i>pre-defined error</i>) do
5 :	moving image multi-resolution and edge creation;
6 :	create features for each points of images in 5;
7 :	normalize features of 3 & 6 ;
8 :	define base point set;
9 :	find correspondent points of defined base points;
10:	if (<i>distance(base point , correspondent point)</i> < <i>desired neighbor-distance</i>)
11:	keep base points and correspondent points;
12:	else
13:	delete base points and correspondent points;
14:	end if;
15:	apply B-Spline transform by remained points to moving image;
16:	moving image \leftarrow registered image
17:	end while;
18:	show the results;

همانگونه که پیشتر گفته شد با استخراج نقاط جدید و اعمال تبدیل بر اساس نقاط جدید متناظر حاصل از مقایسه شباهت در هر مرحله تصویر متحرک به تصویر ایستا نزدیک تر می شود. در شکل های زیر گام های این تبدیل و شباهت تصویر نهایی با تصویر مطلوب به نمایش در آمده است.

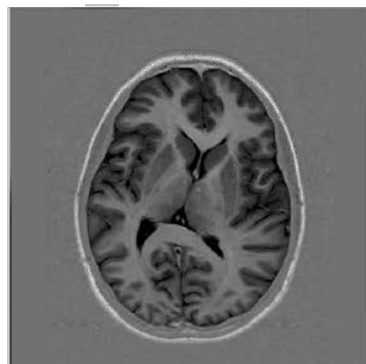
Moving Image



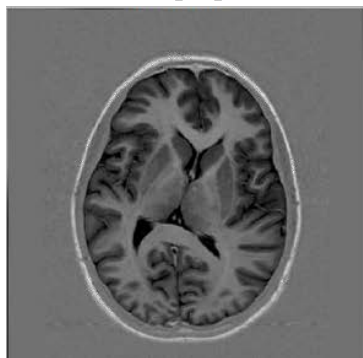
Static Image



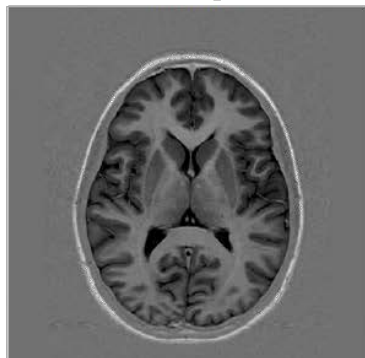
Registered Image



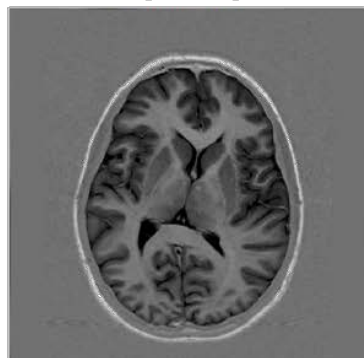
Moving Image



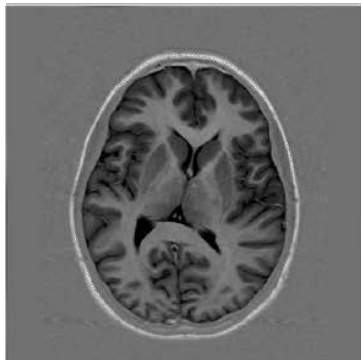
Static Image



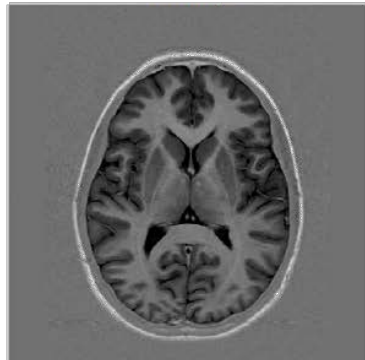
Registered Image



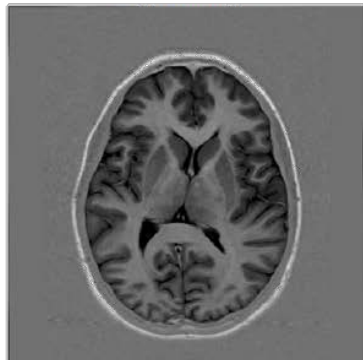
Moving Image

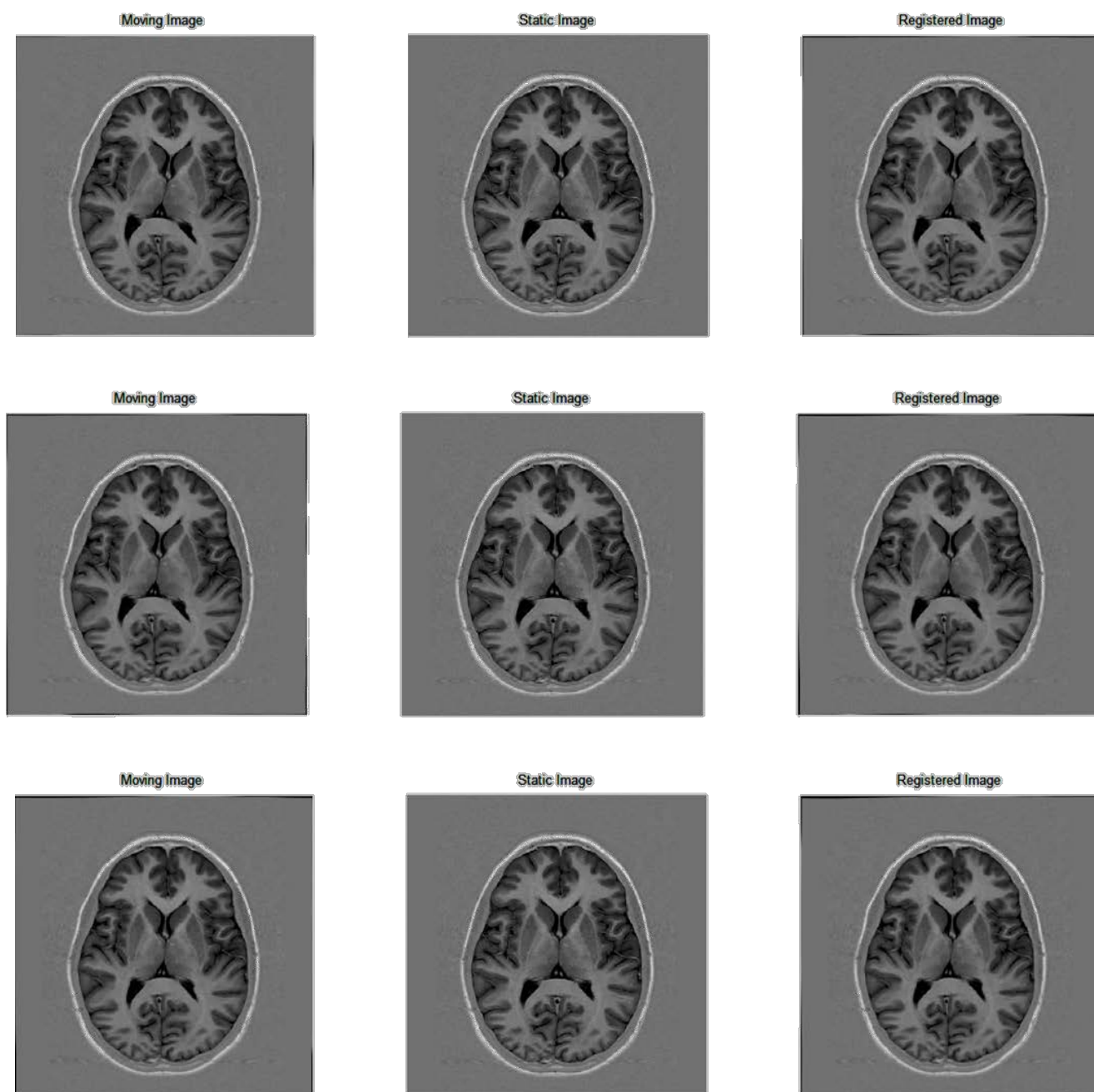


Static Image



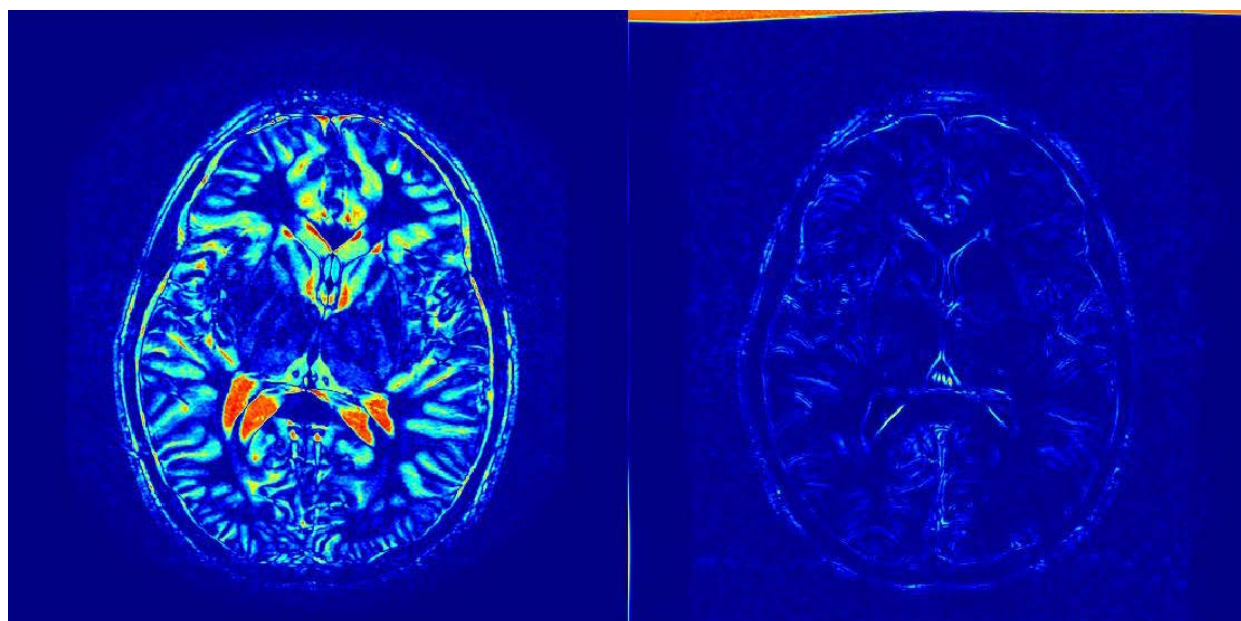
Registered Image





شکل ۶: گام از اجرای حلقه در انطباق تصاویر مبدا و مقصد

همان‌گونه که در تصاویر نیز مشخص است تصویر مقصد در هر گام به تصویر مبدأ نزدیک‌تر می‌شود به گونه ای که در ۶ گام دو تصویر به گونه ای که در شکل ۷ نیز تفاوت تصاویر اولیه و تصویر رجیستر شده به نمایش درآمده نشان می‌دهند به خوبی بر هم منطبق شده‌اند. در شکل (۷) نقاط قرمز رنگ نشان دهنده بیشترین تفاوت دو تصویر و نقاط آبی نشان دهنده تفاوت اندک میان تصاویر می باشد. لازم به ذکر است انتظار می رود تا با ادامه حلقه میزان خطا هر چه بیشتر کاهش یابد.



شکل ۷: اختلاف تصاویر قبل (راست) و بعد (چپ) از انطباق

حال اگر خطا را به مجموع مربعات اختلاف دو تصویر در نظر بگیریم ونقاط حاشیه ای تصویر را که خطایی ناشی از جایی تصویر ایجاد می کنند از این تعریف مستثنی کنیم و مقادیر حاصل را رمالیزه کنیم برای تصویر مبدأ و مقصد این خطا برابر با ۰,۰۱۵ بوده است که پس از اعمال انطباق در گام ششم این مقدار به ۰,۰۰۱۰ رسیده است.

بحث و بررسی

در مقاله حاضر به هدف اصلاح اغتشاشات شامل جابجایی‌های ناهمگون ناشی از حرکات ارگان عکس‌برداری شده، روشی مبتنی بر شناسایی خودکار نقاط نشانه و حذف برخی از این نقاط نشانه به هدف هموارسازی تبدیل صورت گرفته، ارائه شده است. مزیت عمده روش را می‌توان استخراج ویژگی‌های تصویر از هیستوگرام‌های محلی در ترکیب با استفاده از تبدیل غیرصلب B-Spline به همراه روش‌های عملی برای هموارسازی تبدیل عنوان کرد که توانسته با سرعت قابل قبولی این اغتشاشات را اصلاح کند. توانایی عمده روش را می‌توان در کاهش اغتشاشات ناشی از حرکت محلی ناهمگون دسته‌ای از پیکسل‌ها عنوان کرد. نتایج اعمال روش بر روی تصاویر حاصل از دو بیمار مختلف با ساختارهای غیریکسان رضایت بخش نبوده است بنابراین می‌توان در کارهای آینده، روش را برای اعمال به تصاویر حاصل از بیماران مختلف اصلاح کرد.

1. D. Shen, "Image registration by local histogram matching," Pattern Recognition. , vol. 40, pp. 1161–1172, 2007.
2. Dirk-Jan Kroon, "Non-rigid B-Spline grid image registration" ,26 May 2008 (Updated 11 Jan 2010), Available:
www.mathworks.com/matlabcentral/fileexchange/20057-b-spline-grid-image-and-point-based-registration