Registro Rígido Monomodal em Imagens Médicas 2D utilizando Redes Convolucionais

Felippe Trigueiro Angelo
{trigueiro.angelo@gmail.com}
Aluno de Mestrado da Faculdade de Engenharia Elétrica - UNICAMP

Abstract – O presente trabalho irá descrever uma abordagem para a implementação de um algoritmo de registro de imagens utilizando redes convolucionais. O problema de registro de imagens consiste em: dadas duas imagens, onde uma será a referência e a outra a flutuante, encontrar uma transformação geométrica que alinhe a imagem flutuante com a referência. Ao contrário da abordagem tradicional, que trata o problema de registro como um problema de otimização, ele será modelado como uma rede convolucional. A arquitetura de rede convolucional utilizada no presente trabalho foi extraída do trabalho de SLOAN et al. 2018 [13]. Como base de dados (treinamento, validação e teste) foram utilizadas pares de imagens onde a imagem flutuante foi obtida sinteticamente. O resultado da rede convolucional foi comparado com a abordagem clássica de otimização e obteve desempenho superior.

Keywords – Redes Convolucionais, Registro de Imagens Médicas, Deep Learning, Machine Learning, Imagens Sintéticas.

1. Introdução

O algoritmo de registro de imagens consiste em alinhar geometricamente duas imagens. Este alinhamento pode se dar por transformações geométricas rígidas, afins ou deformáveis [3]. As transformações rígidas, onde são aplicadas apenas rotações e deslocamentos, são as transformações em que as propriedades geométricas intrínsecas (área, curvatura, etc) são preservadas. Nas transformações afins, há a inclusão não só de rotações e deslocamentos, mas também de cisalhamento e a mudança de escala. Nelas o paralelismo é preservado. Chamamos o restante das transformações de deformáveis [14], com um número de graus de liberdade maior que as outras, podendo chegar normalmente de dezenas até milhares. A imagem que se deseja deformar é comumente chamada na literatura de imagem flutuante [12], imagem em movimento (*moving volume*) [1] ou imagem deformável [8]. A imagem que servirá como referência para a deformação é comumente chamada na literatura de imagem de referência [12] ou imagem alvo (*target image*) [2].

Em geral na abordagem tradicional de registro, esses algoritmos são constituídos por três etapas [6] (Fig. 1): (i) Modelo de transformação, que corresponde ao tipo de transformação que a imagem deverá ter, podendo ser rígida, afim ou deformável; (ii) Otimizador, que corresponde ao algoritmo utilizado para encontrar o ponto que maximiza ou minimiza a função, onde se destaca o Gradiente Descendente como um algoritmo mais utilizado para o nosso domínio de problemas [11] [12] [2]; e (iii) Função objetivo, que consiste na métrica utilizada para medir o grau de similaridade ou dissimilaridade entre as duas imagens, dentre as quais se destacam as medidas SSD (*Sum of Square Differences*), MI (*Mutual Information*) e CC (*Correlation Coefficient*), que ainda podem ser subdivididas em monomodal, que corresponde ao registro de imagens que foram obtidas sob a mesma modalidade de exame, e multimodal, que corresponde ao registro de imagens que foram obtidas por modalidades diferentes.

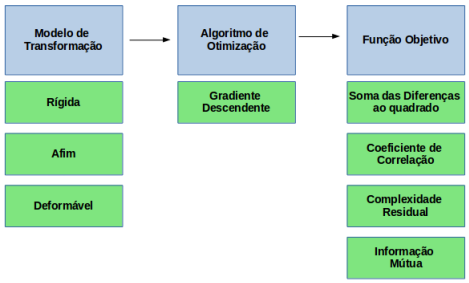

Figura 1. Fluxograma de um algoritmo de registro

Entretanto nos últimos anos o algoritmo de registro de imagens começou a ser também modelado por redes neurais profundas [4] (de agora em diante será utilizado o termo *Deep Learning* sempre que nos referirmos a abordagem e não a rede em si). Na abordagem tradicional que trata o registro como um problema de otimização, alguns efeitos indesejados podem acontecer, como por exemplo: (i) Dificuldade na seleção da função objetivo [10]; (ii) Alto custo computacional [1] e (iii) estagnação do algoritmo de otimização em mínimos locais devido a uma possível não convexidade da função objetivo [13]. Na abordagem utilizando deep learning, é possível se modelar o problema de tal forma que os parâmetros da transformação sejam calculados de forma direta, sem a necessidade de se encontrar o mínimo de uma função objetivo [4], evitando assim os mínimos locais.

Dentre as redes profundas mais utilizadas nesse problema estão as redes convolucionais [4], onde camadas da rede são modeladas por meio da operação de convolução. Assim, com base em um conjunto de dados, os parâmetros do filtro utilizado na convolução são aprendidos.

Com base nas vantagens apresentadas, decidiu-se realizar uma implementação do algoritmo de registro utilizando redes neurais convolucionais. Tal implementação faz parte da nota final da disciplina IA006 - Tópicos em Sistemas Inteligentes II - Aprendizado de Máquina, FEEC/UNICAMP. A arquitetura utilizada foi aquela apre-

sentada por SLOAN et al. [13] para registro monomodal 2D utilizando redes convolucionais.

2. Conjunto de Dados

A base de dados utilizada para treinamento, validação e teste do modelo citado, foi obtida com base nas imagens do conjunto OASIS-1 [9], que consiste em um conjunto de dados livre, contendo imagens de ressonância magnética com contraste T1 correspondendo a 416 indivíduos com idade entre 18 e 91 anos, onde para cada indivíduo tem-se entre 3 e 4 imagens. Cada volume contém 256x256x128 voxels com espaçamento de 1x1x1.25mm.

Para a criação de uma base de dados visando o problema de registro de imagens 2D, necessita-se de um conjunto de pares de imagens. Assim, para cada indivíduo da base de dados OASIS-1 foram selecionadas 121 fatias (uma imagem 3D, como é o caso das imagens do OASIS-1 é formada por um conjunto de fatias 2D) na orientação Sagital. É importante citar que buscou-se evitar a selação de fatias cujo conteúdo era composto primariamente por ar. Para cada uma dessas fatias foram calculados os parâmetros de uma transformação rígida 2D, translação nos eixos X e Y e rotação em torno do centro da imagem. Esses parâmetros foram obtidos aleatoriamente por meio de uma distribuição uniforme, onde as variáveis X e Y estavam na faixa entre -30 e 30, e o ângulo da rotação entre -15 e 15 graus. Após a aplicação da transformação rígida, as imagens tiveram seus valores de pixels normalizados entre 0 e 1. Também foi aplicado um ruído gaussiano com média 0 e desvio padrão 0.1.

Assim, após esse procedimento, foram obtidas 50336 pares de imagens, onde a imagem selecionada como fatia corresponde a Referência e a imagem transformada corresponde a imagem Flutuante. Para o treinamento foram selecionadas 60% dos pares e o restante foi dividido igualmente entre os conjuntos de validação e de teste.

Para a leitura e manipulação das imagens foi utilizada a biblioteca SimpleITK [7] que corresponde a uma biblioteca de manipulação de imagens médicas 2D e 3D e que está disponível para várias linguagens de programação incluindo Python e C++.

3. Arquitetura da Rede Convolucional

Como dito anteriormente na Seção 1., a arquitetura da rede convolucional utilizada na implementação do registro de imagens foi proposta por SLOAN et al. [13] e permite a realização do algoritmo de registro monomodal com transformação rígida. Na implementação dessa rede foi feita uma pequena alteração que consiste na retirada do "Skip Connection"que o autor incluiu para observar o resultado de algumas camadas intermediárias. Assim, a rede utilizada pode ser vista na Figura 2.

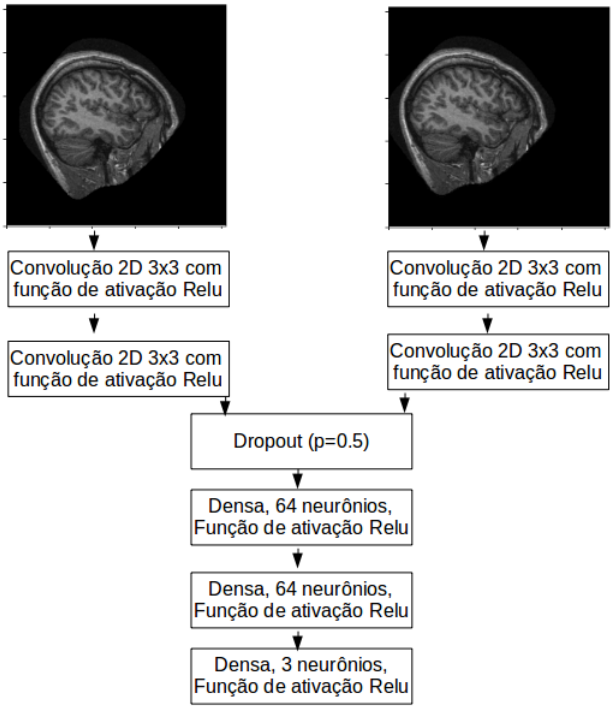

Figura 2. Arquitetura utilizada na implementação do algoritmo de registro.

A rede contém duas entradas que recebem duas imagens de tamanho 256x256. Cada uma delas é inserida em um conjunto de duas camadas convolucionais organizadas em série. Ambas as camadas convolucionais possuem tamanho de filtro 3x3 e função de ativação Relu. Essa série de camadas convolucionais são responsáveis por selecionar algumas regiões na imagem que serão úteis no cálculo dos parâmetros do registro. Assim, como deseja-se selecionar as mesmas estruturas em ambas as imagens, e elas correspondem ao mesmo tipo de imagem (registro monomodal), foram utilizadas camadas convolucionais compartilhadas, onde as camadas que estão no mesmo nível terão os mesmos parâmetros.

Após a aplicação das 2 convoluções em série, as saídas de ambas as camadas são concatenadas ao longo do canal das imagens e posteriormente são convertidas em um vetor unidimensional. Devido ao seu tamanho, a rede construída possui um número elevado de parâmetros o que pode resultar em problemas como o *overfitting*. Assim, foi utilizado A construção da rede, bem como todos os aspectos envolvidos no seu treinamento, validação, teste e análise dos resultados foi feito utilizando a linguagem Python juntamente com a biblioteca Keras, que corresponde a um API voltada para redes neurais.

A técnica Dropout [5] com um fator p de 50%. Por fim, o vetor 1D é inserido em 3 camadas densas em sé-

rie. As duas primeiras são idênticas e possuem 64 neurônios e função de ativação Relu. A última, possui 3 neurônios e função de ativação linear e é responsável por retornar os parâmetros preditos da transformação afim na sequência (X, Y, D).

4. Treinamento

Como dito na seção 2., na etapa de treinamento foram utilizados 30202 pares de imagens. Como otimizador utilizado para treinamento da rede foi utilizado o Adadelta com parâmetros $\rho = 0.5$ e *learning rate*=1. Para a função objetivo foi utilizada a média da soma dos erros quadráticos. Tal escolha se mostra coerente pois os parâmetros envolvidos na transformação correspondem a uma transformação linear. A Figura 3 mostra a curva de treinamento para os conjuntos de treinamento e de teste. Um ponto importante de citar é o fato de em muitas épocas o valor do erro no conjunto de treinamento é maior do que o do conjunto de validação. Tal resultado decorre do fato de que o erro no conjunto de treinamento é calculado como sendo a média do erro em cada iteração, ao contrário do cálculo do erro no conjunto de validação que é calculado no final da época.

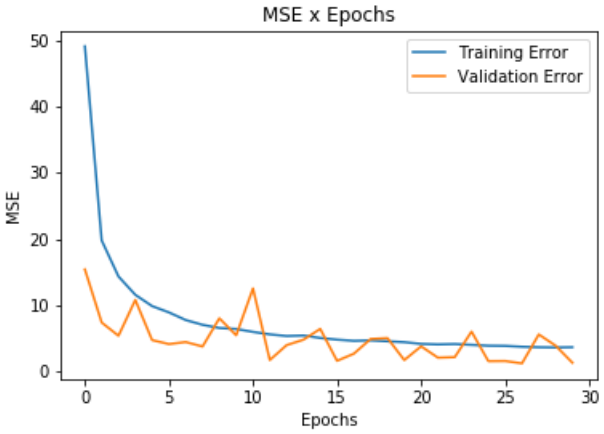

Figura 3. Curva de aprendizado para os conjuntos de treinamento e validação.

A construção da rede, bem como todos os aspectos envolvidos no seu treinamento, validação, teste e análise dos resultados foi feito utilizando a linguagem Python juntamente com a biblioteca Keras, que corresponde a um API voltada para redes neurais.

Devido o conjunto de dados de treinamento ser maior do que a quantidade de memória RAM disponível, o treinamento foi feito utilizando a abordagem online. Assim, o conjunto de treinamento foi dividido em 32 batches de 943 pares de imagem cada. O tempo total de treinamento foi de aproximadamente 10h, em um computador i5-7200U 2.5GHz e memória RAM de 8Gb. Todo esse processamento foi realizado na CPU.

5. Resultados

Após a etapa de treinamento obteve-se um erro de 3.6338 no conjunto de treinamento e de 1.2861 no conjunto de validação. O mesmo modelo também foi aplicado no conjunto de teste, resultando em um erro de 3.8896. Também foram calculadas as médias e o desvio padrão dos erros para cada um dos parâmetros de saída X, Y e D.

Para realizar uma avaliação do modelo obtido, foi realizado o registro dos pares que constituem o conjunto de teste, utilizando a abordagem clássica de registro que o trata como um problema de otimização. Para tal, foi utilizado a biblioteca SimpleITK utilizando o gradiente descendente estocástico como algoritmo de otimização. Nesse algoritmo foram utilizadas 95% das amostras em cada uma das iterações, *learning rate* de 1 e número máximo de iterações de 200. Como função objetivo foi utilizada a Informação Mútua de Mattes com uma quantidade de bins de 64 em cada coordenada. Ao final de cada iteração uma nova imagem transformada é gerada, assim é necessário utilizar um algoritmo de interpolação. Para esse caso, foi utilizada a interpolação linear devido ao seu baixo custo computacional e de não gerar valores fora da faixa dos dados de entrada.

Assim, ao final do registro com o SimpleITK foram obtidas as médias e desvio padrão dos erros para cada um dos parâmetros de saída X, Y e D. O resultado desse algoritmo, bem como a comparação com o resultado da rede convolucional, pode ser visto na Tabela 1. Também foi calculada a média das somas dos erros quadráticos no conjunto de teste, obtendo um resultado de 3079.4, que é 1000X maior que a abordagem utilizando redes convolucionais.

	X	Y	D
CNN	-1.69∓1.09	-0.45∓0.56	-0.42∓0.91
$SimpleITK$	30.52∓25.56	-39.36∓41.50	34.15∓18.85

Tabela 1. Exemplo de uma tabela.

Também é importante citar que o tempo de cálculo do registro no conjunto de teste em ambas as abordagens foram bastante discrepantes. Na abordagem utilizando a CNN o tempo foi de aproximadamente de 5 min ao passo que na abordagem de otimização foi de aproximadamente 1h.

A Figura 4 mostra um mapa de dispersão entre a predição das variáveis de transformação X, Y, D versus a variável original inserida na transformação. No caso ideal, obteríamos uma linha representando uma função identidade, o que não ocorre no caso real. No resultado obtido no experimento o que pode ser visto corresponde a uma função identidade com alguma variância, onde nos

parâmetros da translação tem-se uma variância menor do que no parâmetro correspondente à rotação em graus.

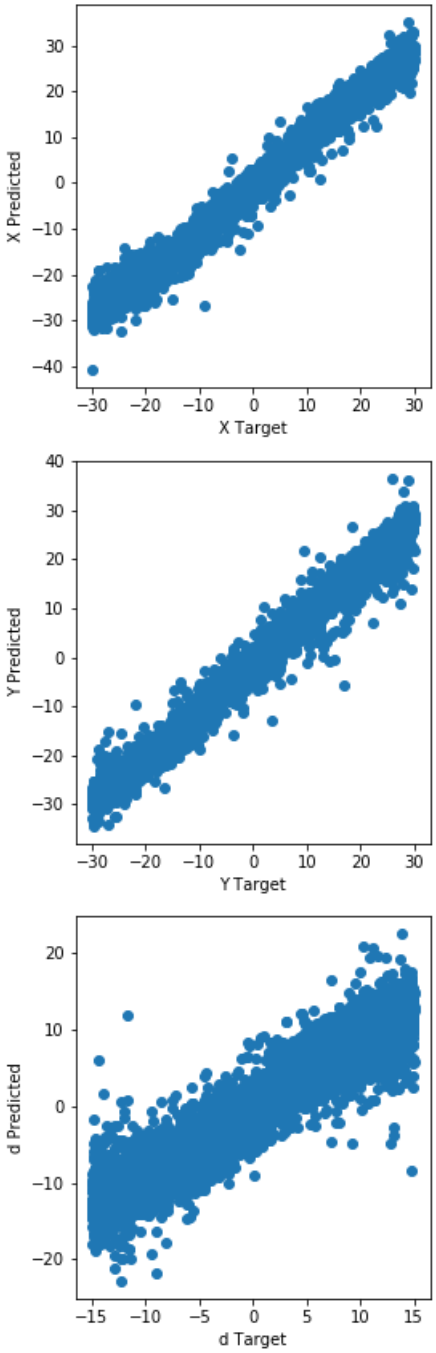

Figura 4. Distribuição do resultado obtido conjunto de testes utilizando CNN versus o parâmetro original da transformação.

Para demonstrar o resultado do registro com a CNN foi aplicado o modelo obtido a um par de imagens do conjunto de testes. A transformação original aplicada na imagem de Referência foi de $X = -15.91$, $Y = 2.21$ e $D = -3.83$. O resultado obtido como predição pela rede neural foi de $X = -16.26$, $Y = 1.47$ e $D = -4.87$, o que corresponde a um baixo erro. O conjunto de imagens 5- 7 demonstra essa abordagem, onde a Figura 7 corresponde a imagem obtida como resultado da transformação com os parâmetros preditos.

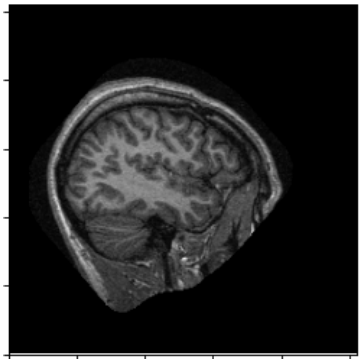

Figura 5. Imagem de Referência.

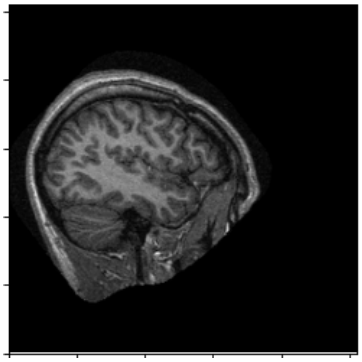

Figura 6. Imagem Flutuante.

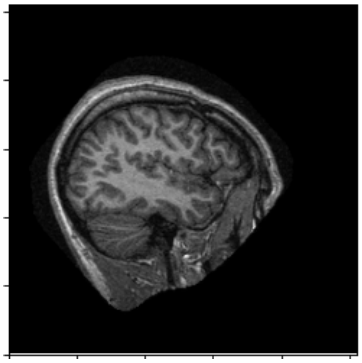

Figura 7. Imagem obtida na transformação predita.

6. Conclusões

No presente trabalho foi implementada uma rede neural convolucional para a solução do problema de registro monomodal de imagens médicas 2D. Para tal foi utilizada a arquitetura sugerida por SLOAN et al. [13]. Como base de dados, os pares utilizados para treinamento, validação e teste foram obtidos do conjunto de dados OASIS-1 [9]. Diferentemente de [13], foram utilizadas ao invés de 500 imagens de validação e teste, 10068 pares de imagens o resulta em medições de erro mais confiáveis. Uma outra diferença foi que não foi utilizado o *Skip Connection* existente na proposta;

Os resultados obtidos com o modelo da CNN foram comparados com a abordagem tradicional de registro que modela o problema como um problema de otimização. Como pôde ser visto, a rede convolucional utilizada obteve um desempenho muito superior à abordagem tradicional tanto na análise do erro quanto no tempo de processamento. Entretanto uma desvantagem das redes neurais consiste na necessidade de se ter uma etapa de treinamento o que é algo com um custo computacional consideravelmente elevado, porém com a evolução dos processadores e GPUs esse é um problema que pode ser menor no futuro.

Um ponto importante de ser citado é que apesar de ter performado muito bem, a CNN não foi testada em um conjunto de imagens diferentes. De fato, em seu artigo, SLOAN et al. [13] mostra que no caso de uma base de dados diferente, o parâmetro da rotação performa pior do que a abordagem tradicional de otimização. Entretanto, esse é um problema que pode ser evitando criando-se um conjunto de treinamento com uma maior variabilidade de máquinas de aquisição de imagens.

Por fim, um último ponto a ser citado é de que mesmo com os mesmos parâmetros e bases de dados desse trabalho é possível se obter resultados consideravelmente diferentes. De fato, durante a implementação deste trabalho foram obtidos erros significativamente maiores. Isso se deve ao fato da aleatoriedade na inicialização do treinamento, assim para obter um bom resultado talvez seja necessária a realização de múltiplos experimentos.

Como perspectivas futuras é possível estender o presente trabalho para que ele possa realizar o registro de imagens 3D. Outra extensão possível também é a de permitir que o algoritmo de registro possa abordar imagens multimodalidade, permitindo por exemplo o registro com atlas médico. Tal extensão também foi proposta por SLOAN et al. [13]. Segundo [4] outros modelos de aprendizagem de máquina estão sendo utilizados (e.g. *reinforcement learning*), assim também é possível estender o presente trabalho para outros modelos e suas variações.

Referências

[1] Xiaogang Du, Jianwu Dang, Yangping Wang, Song Wang, and Tao Lei. A parallel nonrigid registration algorithm based on b-spline for medical images. *Computational and mathematical methods in medicine*, 2016, 2016.

[2] Fatma El-Zahraa, Ahmed El-Gamal, Mohammed Elmogy, and Ahmed Atwan. Current trends in medical image registration and fusion. *Egyptian Informatics Journal*, 17(1):99–124, 2016.

[3] James D Foley, Foley Dan Van, Andries Van Dam, Steven K Feiner, John F Hughes, J Hughes, and Edward Angel. *Computer graphics: principles and practice*, volume 2. Addison-Wesley Professional, 1996.

[4] Grant Haskins, Uwe Kruger, and Pingkun Yan. Deep learning in medical image registration: A survey. *arXiv preprint arXiv:1903.02026*, 2019.

[5] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[6] Juan Eugenio Iglesias and Mert R. Sabuncu. Multi-atlas segmentation of biomedical images: A survey. *Medical Image Analysis*, 24(1):205–219, June/July 2015.

[7] Bradley Christopher Lowekamp, David T Chen, Luis Ibáñez, and Daniel Blezek. The design of simpleitk. *Frontiers in neuroinformatics*, 7:45, 2013.

[8] Frederik Maes, Emiliano D'Agostino, Dirk Loeckx, Jeroen Wouters, Dirk Vandermeulen, and Paul Suetens. Non-rigid image registration using mutual information. In *Compstat 2006-Proceedings in Computational Statistics*, pages 91–103. Springer, 2006.

[9] Daniel S Marcus, Tracy H Wang, Jamie Parker, John G Csernansky, John C Morris, and Randy L Buckner. Open access series of imaging studies (oasis): cross-sectional mri data in young, middle aged, nondemented, and demented older adults. *Journal of cognitive neuroscience*, 19(9):1498–1507, 2007.

[10] Alexis Roche, Gregoire Malandain, and Nicholas Ayache. Unifying maximum likelihood approaches in medical image registration. *International Journal of Imaging Systems and Technology*, 11(1):71–80, 2000.

[11] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes. Nonrigid registration using free-form deformations: Application to breast mr images. *IEEE TRANSACTIONS ON MEDICAL IMAGING*, 18(8):712–721, 1999.

[12] Michaël Sdika. A fast nonrigid image registration with constraints on the jacobian using large scale constrained optimization. *IEEE TRANSACTIONS ON MEDICAL IMAGING*, 27(2):271–281, feb 2008.

[13] James M Sloan, Keith A Goatman, and J Paul Siebert. Learning rigid image registration-utilizing convolutional neural networks for medical image registration. 2018.

[14] Aristeidis Sotiras, Christos Davatzikos, and Nikos Paragios. Deformable medical image registration: A survey. *IEEE TRANSACTIONS ON MEDICAL IMAGING*, 32(7):1153–1190, 2013.