

How to Cite:

Nisha, M., Kannan, T., & Sivasankari, K. (2022). Deep integration model: A robust autonomous segmentation technique for hippocampus in MRI images of human head. *International Journal of Health Sciences*, 6(S2), 13745–13758.
<https://doi.org/10.53730/ijhs.v6nS2.8756>

Deep integration model: A robust autonomous segmentation technique for hippocampus in MRI images of human head

M. Nisha

Assistant Professor, Department of CSE, Akshaya College of Engineering and Technology, Coimbatore

T. Kannan

Professor, Department of Mechanical Engineering, SVS College of Engineering, Coimbatore

K. Sivasankari

Professor, Department of ECE, Akshaya College of Engineering and Technology, Coimbatore

Abstract---Segmentation of hippocampus in Magnetic Resonance Images (MRI) is a vital process in neuro imaging studies towards analyzing and extracting the structure of hippocampus. In existing, many automatic segmentation algorithms such as region based segmentation; boundary based Segmentation has been applied to segment the Hippocampus (HC) as it produces the higher accuracy and less mean squared error. Despite several advantages, the model is time consuming and it generates less accuracy on the volumetric changes of the diverse dataset. However, due to the existence of imaging artifacts, changes in anatomy, contrast variability, and poor registration, most of these mechanisms do not produce suitable results. In order to resolve those issues, a novel framework named as Deep Integration Model has been proposed using Sparse Principle Component Analysis, Affinity Propagation and Linear Convolution Neural Network to segment the MRI Images. The model segments the Hippocampus and its subfields in the MRI images. Affinity Propagation identifies the modalities and pathologically altered tissues and represented it as features set. Sparse Principle Component Analysis extracts the sparse information and its interrelationship on the feature set which is represented as max pooled data. Further Linear Convolution Neural Network has been employed to generate the class labels for the sparse information in the form of feature set. The edges of the class labels on the batch normalization are smoothened using Linear CNN to separate intracranial tissues and skull encircling.

ReLU activation unit generates the class label for the max pooled data. Experimental analysis using MRI dataset on the proposed model explains the improved performance in terms of dice similarity coefficient, Jaccard Coefficient and segmentation accuracy respectively towards HC stripping.

Keywords---hippocampus segmentation, MRI data, linear convolution neural network, sparse principle component analysis, affinity propagation.

Introduction

Hippocampus Segmentation is an important technique in the analysis of neuro imaging data. Cortical surface reconstruction, brain morphometry and pre surgical planning are some of the applications that, depend on the ability to precisely segment brain from non-brain tissue [1][2]. It is an important pre-processing step to analyze the complex structure of the brain. Many automatic approaches such as region-based, boundary based, and hybrid approaches have been applied in existing HC segmentation[3].

Region-based methods detect the connected regions by some predefined criteria like intensity, thresholding, clustering, and filtering for identifying the targeted volume. Boundary-based methods[4] mainly focus on gradient facts in order to identify the surface of hippocampus, usually modeled by an active contour[5][6]. Hybrid approach uses active contour segmentation and morphological operations to identify the hippocampus components [7]. Automatic segmentation of hippocampus is a tedious task because of the low contrast MRIs, ambiguous brain margins and lack of intensity normalization. Furthermore, entire HC segmentation becomes more challenging, when pathological disorder MRI datasets are used [8].

In this paper, novel framework named as Deep Integration Model has been modeled to automate the hippocampus segmentation and smoothen the edges of the class labels. The model uses Sparse Principle Component Analysis, Affinity propagation and Linear Convolution Neural Network. The learning rate is modified to fine-tune each step and the output layer is removed by the AlexNet model implemented by CNN that acts as feature generator. The model segments the Hippocampus subfields in the MRI images. It extracts the HC volume which has the shape of a HC within a certain tolerance. Further it has been proven robustness to noise and could lead to a good primary extraction of the hippocampus surface. The rest of the section is organized as follows, section 2 describes the related work followed by section 3 to define the proposed methodology and section 4 discusses the experimental result and finally section 5 concludes the paper.

Related Works

In this section, various existing model applied to skull stripping using various imaging data has been detailed as follows:

Robust Brain Extraction – Hybrid Approach

It is a well-known learning-based algorithm that uses tree structured classifier capable of extracting and classifying the texture- and intensity-based features of the image dataset that integrates a discriminative and a generative model with the graph cuts for skull stripping. It generates the training model to learn the irregular brain features to precisely segment the brain MRIs [10].

3D U-Net- Deep Network Architecture

It is a well-known semantic segmentation model that uses the deep network architecture for the segmentation of hippocampus in the brain MRI data. The architecture of 3D-UNet in multiscale encompasses a contracting path, an expanding path, and precise localization for the precise use of features. In order to obtain local characteristics the 3D-UNet implements concatenation for combining the up-sampled output with the features of high resolution. [11].

Method

In this section, we define a Deep Integration Model on Brain MRI dataset to segment the skull using Sparse Principle component Analysis, Affinity propagation and Linear Convolution Neural Network using various processing steps

Brain MRI Image Pre-processing

Pre-processing is used to compute set of parameters for consequent processing using Brain extraction tool [12]. The parameters included are the coordinates of the centroid of the brain (COG), an upper bound on the intensity of the cerebrospinal fluid (CSFMAX), and an estimate of the average radius of the brain. This initial estimate is used for distinguishing between brain and non-brain tissues.

Affinity propagation

Affinity propagation (AP) approach is first applied to cluster images pixels represented by exemplars. Possible exemplars are obtained by considering all data points by the reduction of an energy function and message-passing architecture thereby obtaining the corresponding clusters on the optimal set of exemplars. Identify a subset of data points as exemplars and assigns every other data points to one of those exemplar [13]. On assumption, all data points that are identified are simultaneously reflected as exemplars. Exemplars uses the soft max function to differentiate between brain and non-brain tissue. Affinity Propagation identifies the modalities and pathologically altered tissues in the clusters and represented as features set.

Those exemplars that maximize the net similarity are identified by Affinity Propagation. Here member data points and the overall sum of similarities between all exemplars are predicted. Two kinds of messages namely responsibility and availability are exchanged among data points and it can be viewed as a message

passing process. Responsibility, $\text{resp}[i, j]$, is a message from the data point i to j that manipulates the collected evidence for which the data point j suits to serve as the exemplar for the data point i . Availability, $\text{avail}[i, j]$, is a message from the data point j to i that manipulates the collected evidence for which the data point i suits to choose the data point j as its exemplar. Initially all the responsibilities and availabilities are set to 0, and to compute convergence their values are iteratively updated by the algorithm below.

Affinity Propagation Algorithm

Input: A set of pairwise real-valued similarities, $\{\text{sim}(i, k)\}$, between the data points and a real-valued exemplar cost or the number of exemplars (K).

Step 1: A set of pairwise real-valued similarities, $\{\text{sim}(i, k)\}$, where $\text{sim}(i, k)$ is a real number in which it indicates the suitability of the data point k as an exemplar for data point i mentioning tissue

Step 2: $\text{sim}(i, k) = -\|x_i - x_k\|^2, i \neq k$ ---- Responsibility

Step 3: For tissue cluster [21]
 $p[i, j] = \text{sim}[i, j] - \max_k \{a[i, k] + \text{sim}[i, k]\}$

Step 4: For each data point k , a real number, $\text{sim}(k, k)$, indicating the *a priori* preference that it be chosen as an exemplar [29]

Step 5: $\text{sim}[i, j] - \max_k \{j\{\text{sim}[i, k]\}$

Step 6: $\text{sim}(k, k) = p \forall k$ ---- Availability

Step 7: It can be derived as belief propagation (max-product) on a completely-connected factor graph to maximize the sum of similarities between the exemplars and their data points, minus the cost of the exemplar to generate the available feature set.

Output: A subset of the data points on exemplar and a consignment of the data point to an exemplar.

Sparse Principle Component Analysis

Sparse Principle Component Analysis extracts the sparse information and its interrelationship on the feature set which is represented in the form of max pooled data. In this sparse information the clustered feature set represents only a less number of effective features yet it retains most of its intrinsic information in its cluster [14]. Suppose that x of dimension m has to be transmitted using 1

numbers, where $l < m$. If the vector x has been truncated, a mean square error equal to the sum of the variances of the elements has been generated and it will be eliminated from x . Further, there exists an invertible linear transformation T such that the truncation of Tx is optimal in the mean-squared sense. The transformation T should have the property that some of its components have low variance. Sparse Principal Component Analysis maximizes the decrease in the rate of variance.

Sparse Principle Component Analysis Algorithm

Input: Clustered Feature Set

Step 1: Let an m -dimensional random vector say X represent RoI (Region of Interest) in the particular cluster

Step 2: Assume vector X has mean zero $S[X] = 0$ where S is the statistical expectation operator

Step 3: If (X has no zero mean)
($X - \text{Mean}$)

Step 4: Let q denote a unit vector, also of dimension m , onto which the vector X is to be projected

Step 5: $A =$ inner product of the vectors X and q
 $A = X^T q = q^T X$
where A is a random variable

Step 6: Constraint
 $\|q\| = (q^T q)^{1/2} = 1$

Step 7: If (X has zero Mean)
 $S[A] = q^T S[X] = 0$

Step 8: Variance of $A =$
 $\sigma^2 = S[A^2] = S[(q^T X)(X^T q)] = q^T S[XX^T] q = q^T R q$

Step 9: Correlation Matrix $R = S[XX^T]$

Step Matrix has been transformed to $m \times 1$

10: vector a and b as follows

$$a^T R b = b^T R a$$

Step Variance σ^2 of A is a function of the

11: unit vector q

$$\psi(q) = \sigma^2 = q^T R q$$

Above mathematic notation is

represented as a variance probe $\psi(q)$

Step Further feature set containing variance

12: probe undergoes various constraints
are follows

$\psi(q + \delta q) = (q + \delta q)^T R (q + \delta q) - \text{First order Constraint}$

$\psi(q + \delta q) = q^T R q + 2(\delta q)^T R q + (\delta q)^T R \delta q - \text{Second order Constraint}$

Step On applying perturbation δq which is

13: orthogonal to q

$$(\delta q)^T R q - \lambda (\delta q)^T q = 0 \Rightarrow (\delta q)^T (R q - \lambda q) = 0$$

From above constraint, q is an eigen vector and λ is an eigenvalue of R

Step 14: Eigen Value R can be represented as

$$[31] R = \sum_{i=1}^m \lambda_i \bar{q}_i \bar{q}_i^T$$

Step 15: Principle Component of the Cluster contain sparse information given by

$$\alpha_j = q_j^T x = x^T q_j, j=1,2,\dots,m$$

Step 16: In particular, the number of features needed for effective data representation is reduced by eliminating the linear combinations by availing the features with large variance in analysis.

Output Max pooled Feature Set

Linear Convolution Neural Network

Convolution Neural Network (CNN) based algorithms are trained with known labeled data to learn the fundamental mathematical description required for object segmentation and classification [15]. The network architecture consists of four convolutional layers of $3 \times 3 \times 3$ elements for the volume with kernels. Based on the performance of the model on the validation set, the number of convolutional layers is decided. The number of training iteration is defined automatically based on the typical performance on the training set and validation. The Fig. 1 represents the proposed architecture of the work

Max pooling Layer

A $3 \times 3 \times 3$ max-pooling layer is used to compute the maximum value in the feature set as down sampling with stride of 2 in each dimension. The generalization capacity of the model is further increased and the computational volume is also reduced by adding robustness to noise. The number of feature channels is doubled after each down sampling. Feature regularization is carried out to eliminate the over fitting issues.

Batch Normalization

Batch normalization (BN) is used for the faster convergence of the features processed through Sparse PCA. BN is the normalization of the activation function value or the output value of the convolution [1]. It is not influenced by a parameter scale during weight propagation, when batch normalization is used. Thus, the learning rate controls and adjusts the weights during increasing and enabling of rapid learning.

Convolution Layer

The Convolution layers capture the batch normalized features from low level to more abstract features through their integral mechanism hierarchically and learn the discriminative features. Zero-padding around each pixel is made followed by

convolution on the padded image that results in the final convolution filter size of $1 * 1 * 1$.

Activation Function

In order to introduce non-linearity to the system, the architecture uses the Rectified Linear Units (ReLU) activation function. The system generalization is improved by normalizing the output of the activation function where each and every activation function is followed by batch normalization, and overfitting.

Output Layer

The output obtained from the deepest convolutional layer is flattened and given to the fully connected layer, which in turn process the features extracted through convolutional layers and serve as a classifier in the architecture. The most discriminative features are learnt by the system using soft max and cross entropy mechanism. These both mechanisms compares the features with ground truth label collected or generated using manual segmentation process.

Data Visualization

The data visualization has been employed on output layer of Linear CNN towards identifying the discriminative segments of the fully connected features through weighted feature layer to determine the discriminative features processed in linear CNN. The Saliency map has been derived to map the features on interpreting and rationalizing the decision of the trained system towards the interrelationship feature between pixels.

$$x_{ij} = \frac{1}{K} \sum_{l=0}^k x(l)$$

In the above equation, K termed as weight factor and x is considered as feature. The system will detect the hippocampal subfields like CA1, CA2, CA3, DG, SUB, Head and Tail

Linear Convolution Neural Network Algorithm

Input: Feature set $F=\{x_1, x_2, \dots, x_N\}$
Step 1: While feature set $f: 1 \rightarrow F$ do
 Compute the j activation function
 using Rectified Linear Unit (ReLU)
 capture various hippocampal subfields
 in the feature set
Step 2: Generate Noise Vector n
Step 3: Compute Error e
Step 4: Apply Feed Forward Propagation to
 compute cross entropy Gradient $E(\theta)$
Step 5: $cd = \text{Convolution}(F)$
Step 6: $mp = \text{Max_pooling}(cd)$
Step 7: $Bn = \text{Batch Normalization}(mp)$
Step 8: $fc = \text{Fully Connected}(Bn)$

Step 9: Update Network Parameter θ using gradient descent
Output: Target Label $T=\{y_1,y_2..y_N\}$

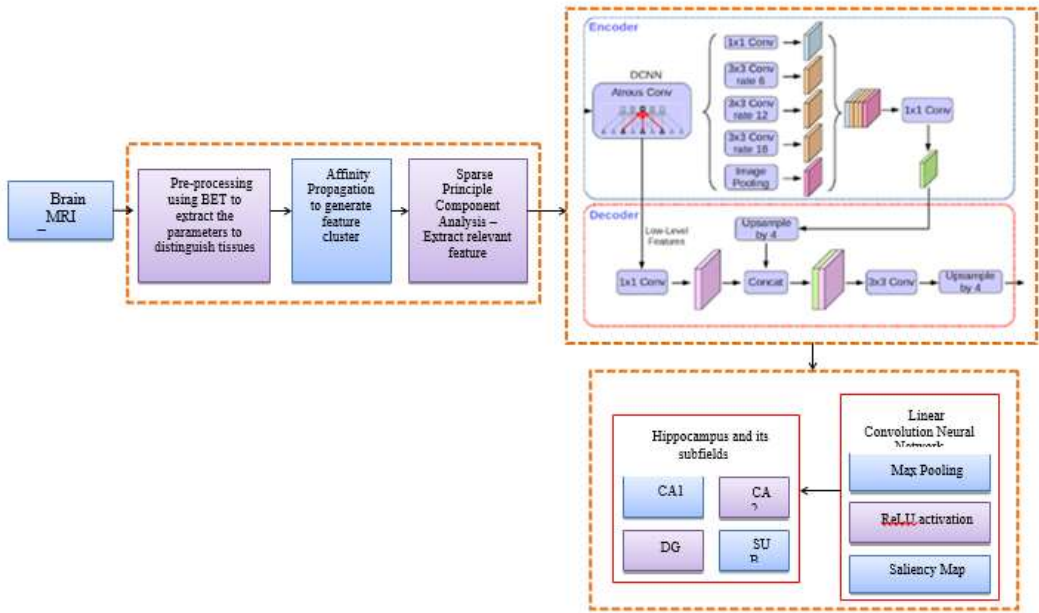


Fig. 1. Proposed Architecture

Automated Hippocampus SegmentationAlgorithm	
Input:	Saliency Map
Step 1:	For D = 1 to N do Generate a training data
Step 2:	For i = 1 to M do Calculate Class label
Step 3:	Class label = Soft_Max (fc)
Step 4:	Class Labels = CA1, CA2, DG, SUB
Step 5:	The system segments (localize) the pathological tissue changes into various categories.
Output:	Segments

Experiment and results

Experimental analysis has been carried out in the brain MRI dataset which is collected from the publically available kaggle dataset. The MRI hippocampus segmentation dataset is collected from kaggle repository. The dataset contains 18900 images with 100 classes and it is created by Saber Malekzadeh. The original size of the images in the dataset is 197x 233 pixels. The original images are resized to 197x197 pixels. The model is implemented in Matlab (version 2017)

for executing the deep learning architecture. Here processing of the image and to train and validate the system is highly challenging. During processing, 60% of the data has been taken for training, 20% of the data has been utilized for validation and remaining, 20% of data has been employed for testing. In this experiment, 10 fold validation has been applied to improve the performance of segmentation. The performance of the model has been evaluated with Dice coefficient, sensitivity, and specificity.

Dice similarity Coefficient

It is computed by considering the difference between the segmented result and ground truth data. In addition, it can be computed using true positive, false positive and false negative values of segmentation results. It is denoted as

$$\text{Dice Similarity Coefficient} = \frac{2TP}{2TP+FP+FN}$$

Sensitivity

It is the measure of percentage of true positive which computes the hippocampus correctly in terms of various features. It is represented as [18]

$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

Specificity

It is the measure of percentage of true negative which computes the non skull correctly in terms of features of the data. It is represented as [18]

$$\text{Specificity} = \frac{TN}{TN+FP}$$

Table 1. Performance Evaluation of Autonomous Segmentation Techniques

Samples	Technique	Dice Coefficient	Sensitivity	Specificity
Test 1 Samples	Deep Integration Model – Proposed model	0.9978	0.9412	0.9989
	3D- U Net- Existing Model	0.9851	0.9336	0.9853
Test 2 Samples	Deep Integration Model – Proposed model	0.9965	0.9514	0.9971
	3D- U Net-	0.9842	0.9389	0.9889

	Existing Model			
Test 3 Samples	Deep Integration Model – Proposed model	0.9956	0.9615	0.9965
	3D- U Net- Existing Model	0.9836	0.9399	0.9841

The analysis on different test samples on various autonomous segmentation techniques has been evaluated on dice coefficient, sensitivity and specificity in the Table 1. It proves that proposed model segments the hippocampal tissues more accurately and excludes the non-hippocampal regions. The proposed deep integration model shows excellent performance compared to existing methods. Without any parameter, tuning the proposed model provides improved performance. The deep learning model is consistent on various cross validation results. The deep integration model produces nearest results on validating with ground truth data.

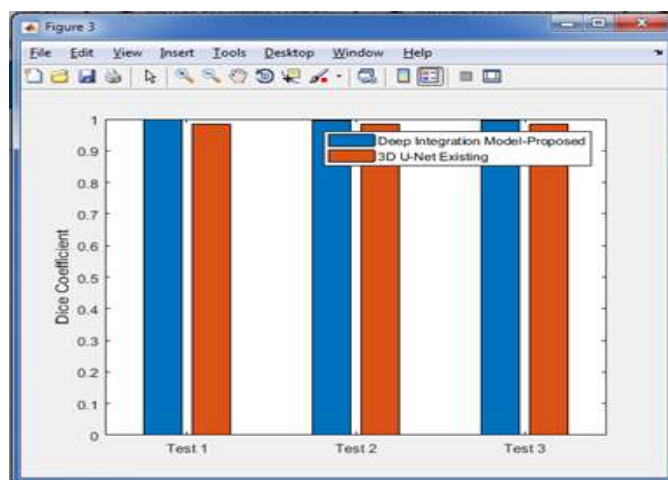


Fig. 3. Performance Comparison of hippocampus segmentation model in terms of Dice Coefficient

Fig. 2. represents the deep integration model segmentation result against brain MRI image and ground truth data. Further the dice similarity coefficient produces excellent results on evaluating with segmentation results of hippocampus subfields such as CA1, CA2, DG, CA3, head and tail in the Fig. 3.

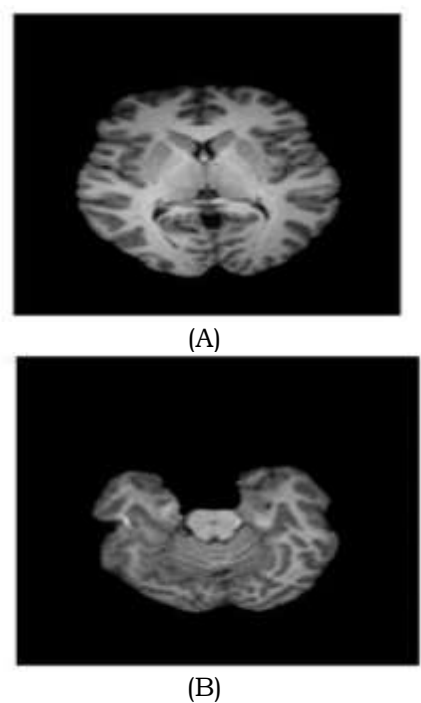


Fig. 2. (A) Input Brain MRI Image (B)Results of segmentation using Deep Integration Model

The sensitivity measure is performed based on the average. In order to tune the sensitivity and specificity by considering the cross validation of the results, threshold is varied while generating the binary masks. Fig. 4. represents the results of sensitivity on the hippocampus subfields.

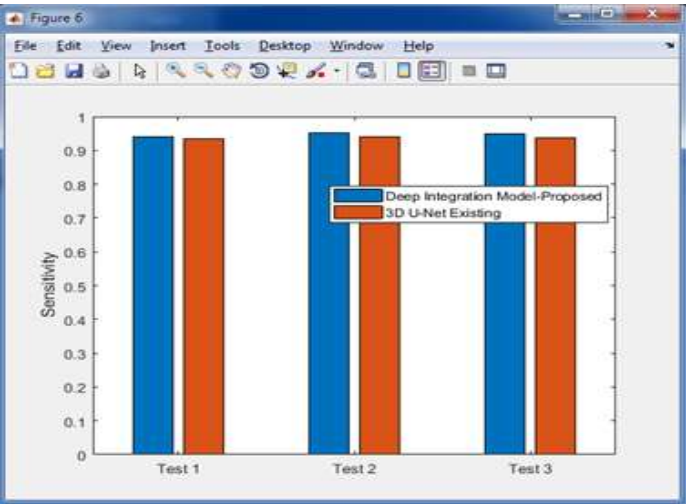


Fig. 4. Performance comparison of hippocampus segmentation model in terms of Sensitivity

The possibility of the performance on trained network has been adapted effectively with target domain, as it results better with specificity value. The Fig. 5. represents the performance comparison of the specificity measure. Instead, suitable features for the given task arise during training automatically.

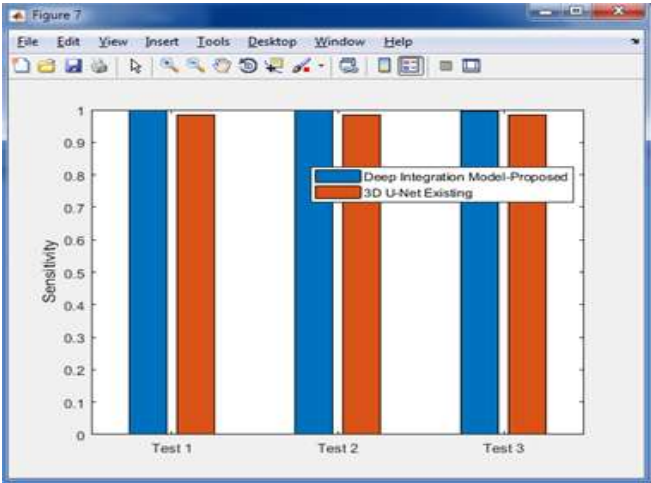


Fig. 5. Performance comparison of hippocampus segmentation model in terms of Specificity

Further, the different autonomous segmentations methods results in various slices of brain MR image on terms of the hippocampus subfields as depicted in Fig. 6. Saliency map identifies the hippocampus Subfields accurately on the segmented results. It is proven that this modified method is really of great significance to speed up the training process.

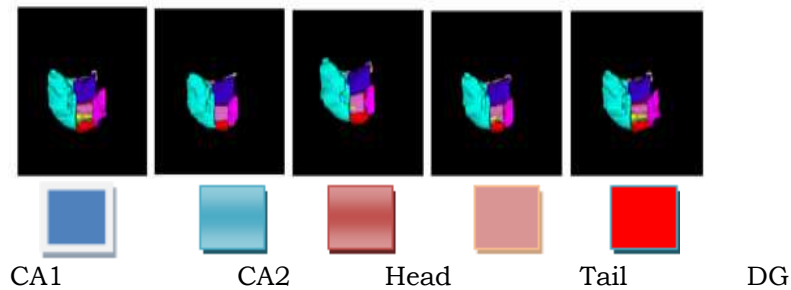


Fig. 6. Hippocampus subfields of Skull

In addition, for maintaining feature maps and spatial dimension of the images through each convolution layer zero padding convolution layers are adopted. The model can even determine the larger pathological changes, like brain tumors or multiple sclerosis.

Conclusion

In this work, design and implementation of novel framework named as Deep Integration Model has been carried out on inclusion of the Sparse Principle

Component Analysis, Affinity propagation and Linear Convolution Neural Network by segmenting the hippocampus and its subfields of the skull in the brain MRI Images. In brain diagnosis, Skull stripping in the Magnetic Resonance Images (MRI) is an essential step in neuroimaging studies in order to exploit various information of the brain. This autonomous segmentation technique based on deep learning produces higher accuracy and less mean squared error with less time consumption. The proposed approach produces better results on imaging artifacts, volumetric changes, varying contrast properties, anatomical variability, and poor registration of the diverse dataset on analyzing the modalities and pathologically altered tissues. The experiments on the publically available dataset confirm that the proposed method provides better results in segmenting hippocampus.

References

1. Uhlich, M.; Greiner, R.; Hoehn, B.; Woghiren, M.; Diaz, I.; Ivanova, T.; Murtha, A. Improved Brain Tumor Segmentation via Registration-Based Brain Extraction. *Forecasting* 2018, 1, 59–69.
2. Zhou, F.; Zhuang, Y.; Gong, H.; Zhan, J.; Grossman, M.; Wang, Z. Resting State Brain Entropy Alterations in Relapsing Remitting Multiple Sclerosis. *PLoS ONE* 2016, 11, e0146080.
3. Iglesias, J.E.; Liu, C.Y.; Thompson, P.M.; Tu, Z. Robust brain extraction across datasets and comparison with publicly available methods. *IEEE Trans. Med. Imaging* 2011, 30, 1617–1634.
4. Kleesiek, J.; Urban, G.; Hubert, A.; Schwarz, D.; Maier-Hein, K.; Bendszus, M.; Biller, A. Deep MRI brain extraction: A 3D convolutional neural network for skull stripping. *Neuroimage* 2016, 129, 460–469.
5. Litjens, G.; Kooi, T.; Bejnordi, B.E.; Setio, A.A.A.; Ciompi, F.; Ghafoorian, M.; van der Laak, J.; van Ginneken, B.; Sanchez, C.I. A survey on deep learning in medical image analysis. *Med. Image Anal.* 2017, 42, 60–88.
6. Leung, K.K.; Barnes, J.; Modat, M.; Ridgway, G.R.; Bartlett, J.W.; Fox, N.C.; Ourselin, S.; Initia, A.D.N. Brain MAPS: An automated, accurate and robust brain extraction technique using a template library. *Neuroimage* 2011, 55, 1091–1108.
7. Eskildsen, S.F.; Coupe, P.; Fonov, V.; Manjon, J.V.; Leung, K.K.; Guizard, N.; Wassef, S.N.; Ostergaard, L.R.; Collins, D.L.; Alzheimer's Disease Neuroimaging, I. BEaST: Brain extraction based on nonlocal segmentation technique. *Neuroimage* 2012, 59, 2362–2373.
8. Heckemann, R.A.; Ledig, C.; Gray, K.R.; Aljabar, P.; Rueckert, D.; Hajnal, J.V.; Hammers, A. Brain Extraction Using Label Propagation and Group Agreement: Pincram. *PLoS ONE* 2015, 10, e0129211.
9. Hariharan, B.; Arbeláez, P.; Girshick, R.; Malik, J. Hypercolumns for object segmentation and fine-grained localization. In *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 7–12 June 2015; pp. 447–456.
10. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation In *Proceedings of the International Conference on Medical image computing and computer-assisted intervention*, Munich, Germany, 5–9 October 2015; pp. 234–241.

11. Milletari, F.; Ahmadi, S.A.; Kroll, C.; Plate, A.; Rozanski, V.; Maiostre, J.; Levin, J.; Dietrich, O.; Ertl-Wagner, B.; Botzel, K.; et al. Hough-CNN: Deep learning for segmentation of deep brain regions in MRI and ultrasound. *Comput. Vis. Image Underst.* 2017
12. Smith, S., 2002. Fast robust automated brain extraction. *Human Brain Mapping* 17, 143 – 155.
13. Salehi, S.S.M.; Erdogmus, D.; Gholipour, A. Auto-Context Convolutional Neural Network (Auto-Net) for Brain Extraction in Magnetic Resonance Imaging. *IEEE Trans. Med. Imag.* 2017, 36, 2319–2330.
14. M. G. Uberti, M. D. Boska, and Y. Liu, “A semi-automatic image segmentation method for extraction of brain volume from in vivo mouse head magnetic resonance imaging using constraint level sets,” *J. Neurosci. Meth.*, vol. 179, no. 2, pp. 338–344, 2009.
15. Çiçek, Ö.; Abdulkadir, A.; Lienkamp, S.S.; Brox, T.; Ronneberger, O. 3D U-Net: Learning dense volumetric segmentation from sparse annotation. In *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, Athens, Greece, 17–21 October 2016; pp. 424–432.
16. Andreas Maier, Christopher Syben, Tobias Lasser, Christian Riess, A gentle introduction to deep learning in medical image processing, *Journal of Medical Physics*, Vol. 29, No. 2, May 2019.
17. Konstantinos Kamnitsas, Christian Ledig, Virginia F.J. Newcombe, Joanna P. Simpson, Andrew D. Kane, David K. Menon, Daniel Rueckert, Ben Glocker, Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation, *Elsevier Journal of Medical Image Analysis*, Vol. 36, pp. 61–78, 2017.
18. Mohammad Hesam, Hesamian, Wenjing Jia, Xiangjian He, Paul Kennedy, Deep Learning Techniques for Medical Image Segmentation: Achievements and Challenges, *Journal of Digital Imaging*, Vol. 32, 2019.
19. Eli Gibson, Wenqi, Carole Sudre, Lucas Fidon, Dzhoshkun I. Shakir, Guotai Wang, Zach Eaton-Rosen, Robert Gray, Tom Doel, Yipeng Hu, Tom Whyntie, Parashkev Nachev, Marc Modat, Dean C. Barratt, Sébastien Ourselin, M. Jorge Cardoso, Tom Vercauteren, NiftyNet: a deep-learning platform for medical imaging, *Elsevier Journal of Computer Methods and Programs in Biomedicine*, Vol. 158, pp. 113–122, 2018.
20. Shengfeng Liu, Yi Wanga, Xin Yang, Baiying Lei, Li Liu, Shawn Xiang Li, Dong Ni, Tianfu Wang, Deep Learning in Medical Ultrasound Analysis: A Review, *Elsevier Journal of Engineering*, Vol. 5, No. 2, pp. 261–275, 2019.
21. Alexander Selvikvåg Lundervold, Arvid Lundervold, An overview of deep learning in medical imaging focusing on MRI, *Journal of Medical Physics*, Vol. 29, No. 2, May 2019.
22. Chao Jia, Dong Liu, Fanshu Kong, Changrun Jia, Three-Dimensional Segmentation of Hippocampus in Brain MRI Images Based on CN-net, 3rd International Conference on Innovation in Artificial Intelligence 2019.