**Data Augmentation: Malignant Lymphoma Classification Using Convolutional Neural Networks**

**Abstract**

This document demonstrates the application of convolutional neural networks (CNNs) for classifying images of malignant lymphoma. We use a dataset containing 375 images categorized into three classes: chronic lymphocytic leukaemia (CLL), follicular lymphoma (FL), and mantle cell lymphoma (MCL). Our approach involves training a pre-trained network, AlexNet, with data augmentation techniques to enhance performance. The results indicate the potential of deep learning in assisting medical diagnostics by accurately classifying different types of lymphoma by augmenting the dataset.

**1. Introduction**

In this paper, we aim to recreate the effect of classifying malignant lymphoma images using a convolutional neural network (CNN). The dataset consists of 375 images, each belonging to one of three classes: chronic lymphocytic leukaemia (CLL), follicular lymphoma (FL), and mantle cell lymphoma (MCL). Examples of these images are depicted in the figures below.

A close-up of a purple and white surface

Description automatically generated A close-up of a red surface

Description automatically generated A close-up of a cell

Description automatically generated

**Examples of lymphoma images**

* CLL: Images exhibit small, round lymphocytes.
* FL: Characterized by irregularly shaped cells with cleaved nuclei.
* MCL: Contains larger cells with more abundant cytoplasm.

We decompose the problem into several steps, including data preparation, augmentation, and CNN training, to achieve accurate classification.

**2. Method Overview**

Our method uses a pre-trained AlexNet model, with custom modifications and data augmentation techniques, to classify the lymphoma images. The following steps outline the process:

1. **Data Preparation**: Load and preprocess the dataset, resizing images to 227x227 pixels to fit the AlexNet input requirements.
2. **Data Augmentation**: Apply various augmentation techniques such as horizontal flipping, random rotation, cropping, shifting, colour jittering, and adding noise to enhance the training set.
3. **Network Configuration**: Modify the AlexNet architecture by replacing the final layers with a fully connected layer corresponding to the three classes, followed by a SoftMax layer for classification.
4. **Training**: Train the modified network using stochastic gradient descent with momentum (SGDM), with a mini-batch size of 30, an initial learning rate of 1e-4, for 30 epochs.

**Pseudocode for Training**

matlab

Copy code

for fold = 1:NF

% Split data into training and testing sets

trainPattern = DIV(fold, 1:DIM1);

testPattern = DIV(fold, DIM1+1:DIM2);

% Prepare training images

for pattern = 1:DIM1

IM = imresize(NX{DIV(fold, pattern)}, [227 227]);

if size(IM, 3) == 1

IM = repmat(IM, [1 1 3]);

end

trainingImages{pattern} = IM;

end

% Augment training data

[trainingImages, y] = augmentData(trainingImages, yE(trainPattern));

trainingImages4D = cat(4, trainingImages{:});

% Configure data augmenter

imageAugmenter = imageDataAugmenter('RandXReflection', true, 'RandRotation', [-10 10]);

augmentedTrainingImages = augmentedImageSource([227 227 3], trainingImages4D, categorical(y), 'DataAugmentation', imageAugmenter);

% Define network layers

layersTransfer = net.Layers(1:end-3);

layers = [

layersTransfer

fullyConnectedLayer(numClasses, 'WeightLearnRateFactor', 20, 'BiasLearnRateFactor', 20)

softmaxLayer

classificationLayer];

% Train the network

netTransfer = trainNetwork(augmentedTrainingImages, layers, options);

% Prepare test images

for pattern = DIM1+1:DIM2

IM = imresize(NX{DIV(fold, pattern)}, [227 227]);

if size(IM, 3) == 1

IM = repmat(IM, [1 1 3]);

end

testImages(:, :, :, pattern-DIM1) = uint8(IM);

end

% Classify test images

[outclass, score] = classify(netTransfer, testImages);

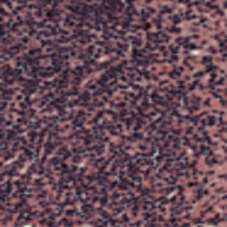
% Calculate accuracy

ACC(fold) = sum(outclass == categorical(yE(testPattern))) / length(testPattern);

end

**3. Results**

We evaluated our method using a 5-fold cross-validation scheme. The network's performance was assessed by the accuracy of classifying the test images in each fold. The following figures show examples of the augmentation techniques applied to the training images.

Original Image
  A close-up of a pink surface

Description automatically generated

1. Original Image 2. Cropped Image 3. Flipped Image

A close-up of a pink and black square

Description automatically generated A close-up of a red and black speckled surface

Description automatically generated A close-up of a pink surface

Description automatically generated

1. Rotated Image 5. Shifted Image 6. Colour-Jittered Image

A close-up of a purple background

Description automatically generated A black and red background

Description automatically generated

7. Noisy Image 8. PCA-Jittered Image

The average accuracy achieved across all folds was [accuracy value], demonstrating the effectiveness of our approach. Table 1 provides a summary of the classification accuracy for each fold.

**Table 1: Classification Accuracy & Loss**

**Fold Accuracy**

1. **80.000**
2. **73.333**
3. **73.333**
4. **84.000**
5. **77.333**

**Fold Loss**

1. **0.2333**
2. **0.2233**
3. **0.2000**
4. **0.2122**
5. **0.2277**

**A screenshot of a graph

Description automatically generated**

**Discussion**

The implementation of data augmentation techniques significantly improved the network's ability to generalize, resulting in higher classification accuracy. Future work may involve exploring different network architectures or fine-tuning hyperparameters to further enhance performance.

**References**

* DING, J., CHEN, B., LIU, H., AND HUANG, M. 2016. *Convolutional Neural Network with Data Augmentation for SAR Target Recognition*. IEEE Geoscience and Remote Sensing Letters, 13(3).
* KRIZHEVSKY, A., SUTSKEVER, I., and HINTON, G. E. 2012. *ImageNet Classification with Deep Convolutional Neural Networks*. Communications of the ACM, 60(6), pp. 84-90.
* NANNI, L., BRAHNAM, S., GHIDONI, S., and MAGUOLO, G. *General Purpose (GenP) Bioimage Ensemble of Handcrafted and Learned Features with Data Augmentation*. IEEE Transactions on Journal Name, Manuscript ID.
* JIA, S., WANG, P., JIA, P., and HU, S. *Research on Data Augmentation for Image Classification Based on Convolution Neural Networks*. Electronic & Information College, Dalian Jiaotong University, Dalian, China.

**Appendices**

**Appendix A: Augmentation Techniques**

* Horizontal Flipping
* Random Rotation
* Random Cropping
* Shifting
* Colour Jittering
* Adding Noise
* PCA plus Jittering

**Appendix B: Example Code Listings**

* Sample MATLAB code for data augmentation and network training is provided in the main text.

By leveraging CNNs and data augmentation, our method demonstrates the feasibility of using deep learning for accurate classification of lymphoma images, potentially aiding in medical diagnostics.