Paper Title	Journal Name and year	Citations	Description
1) DeepPap: Deep Convolutional Networks for Cervical Cell Classification	IEEE journal of biomedical and health informatics 2017	435	**Contribution of work**: Direct classification without prior segmentation using CNN. (pre-trained then fine tuned on cervical dataset) Evaluated on PAP test and LBC datasets. **Database used:** Herlev database, 917 images, 7 classes, HEMLBC dataset obtained from biopsy **Proposed System**: <u>Data preprocessing:</u> <u>Patch extraction:</u> Don't use whole cell images as in the Herlev dataset, Cell segmentation is difficult, hence patches around nucleus are taken for classification. Hence translate nucleus centroid for patch extraction. <u>Data Augmentation</u> Rotate cell images to generate more samples which serves as training data- enhances cell diversity Balanced training data is used since abnormal cells outnumber normal ones in the Herlev dataset. Prevents classifier bias <u>CNN layer:</u> It takes raw pixel intensity images as input and outputs class predictions. Weights optimized using backpropagation. ReLU function boosts training speed by adding non-linearity. Shared filters across feature maps detect patterns in various locations. <u>Pooling layer:</u> Down samples feature map <u>Fully Connected layer:</u> They fuse feature maps into a feature vector for classification. The last fully connected layer computes classification probabilities using softmax regression. <u>Network training:</u> Not understood <u>Transfer learning:</u> Refers to the fine-tuning of deep learning models that are pre-trained on other large-scale image datasets. Initial layers are taken from a pre-trained ConvNet on ImageNet. New task specific layers added. To share features, ConvNet layers are copied from BVLC CaffeNet. Everything is trained together on cervical cell data. **Testing and Eval, Conclusion, future scope:** Not understood, need more time to read

2) Cervical cancer classification using convolutional neural networks and extreme learning machines	Future Generation Computer Systems- Elsevier 2019	220	**Contribution of work**: 1) Introduction of CNNs for cervical cancer cell detection and classification. 2) Introduction of ELM and AE-based classifiers for the same purpose. **Database used:** Herlev dataset with 917 cells and 7 classes (3 normal, 4 abnormal). **Proposed system**: A deep CNN model is trained with a large dataset, then used as a pretrained model which is then fine-tuned with a specific training set and used for testing. Mini batch Stochastic Gradient Descent (SGD) algorithm used to optimize the parameters of the model in all 3. <u>Shallow Architecture:</u> 2 Convolutional layers 2 Max pooling layers ReLU Activation 2 Fully connected layers Softmax Output layer <u>Deep Architecture:</u> Investigated models like VGG 16 Net and CaffeNet. <u>ELM-based Classifier:</u> Two ELMs used after the last fully connected layer of the CNN model. First ELM distinguishes between normal and abnormal cells. Second ELM classifies the normal and abnormal cells into specific classes. <u>AE Classifier:</u> One hidden layer for noise removal and feature extraction. **Results:** Fine tuning: 80% data Testing : 20% data After 5 iterations all data tested. Final accuracy: avg of all 5 iterations. ELM Classifier gave 99.7 % accuracy. Refer to the table in the paper. Deep CNN is better than shallow CNN. **Future scope:** Use other database Incorporating hand crafted features New Deep arch : ResNet, Inception and tree based models.
3) Internet of health things-driven deep learning	The Journal of Supercomputing - Springer	147	**Contribution of work:** 1) Interactive IOHT diagnostic system for cervical cell recognition and classification 2) Deep cervical web app for prediction of normal and

system for detection and classification of cervical cells using transfer learning	2020		abnormal cells. 3) Usage of pretrained convolutional model through web app **Database Used:** Herlev Dataset, Training: 934, Testing: 116 images, classification: abnormal and normal **Proposed System:** Detect and classify PAP Smear images using transfer learning. Combine CNN with ML techniques: a) Knearest neighbor b) Naive Bayes, c) logistic regression d) random forest e) SVM Feature extraction via: a) Inceptionv3 b) VGG19 c) SqueezeNet d) ResNet50 **Convolutional pretrained models:** Used for image pre-processing and feature engineering to recognise cancerous and normal cells. <u>1) Convolution layer: 2 phases</u> Parameters has learnable channels In forward pass convolve each channel over width and height of info vol of img and calculate dot product of kernel and pixel img. In backward pass, gradient of loss wrt to input and bias is calculated <u>2) InceptionV3: 48 deep layers</u> Trained on ImageNet data, known for wider range and limited parameter size, less prone to overfitting <u>3) VGG19: 19 deep layers</u> Image localisation and bounding box structure, Due to extra weights slow training and cumbersome <u>4) SqueezeNet</u> Uses fire modules: Squeeze and expand phases, Squeeze: reduces depth Expand: Increases depth Utilizes deep compression to reduce no of features <u>5) ResNet50: 50 layers</u> Classifies 1000 objects in one iteration, known for high accuracy through ReLU activation. <u>Transfer learning:</u> Context network is trained using a suitable dataset and the task is transferred to a specific target to be trained by the target dataset.

			ML Techniques: Utilized various ML techniques to classify cells as normal or cancerous: Naive Bayes: Assumes independence among features within classes. Random Forest: Constructs multiple decision trees and combines results through voting. SVM: Separates image data using linear or polynomial tricks to maximize class margin. K-NN: Predicts based on nearest neighbors, exploring different neighbor values. Logistic Regression: Makes predictions using a logarithmic function based on training data distribution. **Conclusion:** Highest accuracy: ResNet50 , random forest classifier = 97.89% **Future scope:** Classifying other cells and tissues

1) https://scholar.google.com/citations?view_op=view_citation&hl=en&user=kZn0f6gAAAAJ&citation_for_view=kZn0f6gAAAAJ:hMG6n2O2MHsC

2) https://scholar.google.com/citations?view_op=view_citation&hl=en&user=CVynqVAAAAAJ&citation_for_view=CVynqVAAAAAJ:vV6vV6tmYwMC

3) https://scholar.google.com/citations?view_op=view_citation&hl=en&user=zCalDU8AAAAJ&citation_for_view=zCalDU8AAAAJ:blknAaTinKkC

Extra reading:

Convolutional Neural Network:
Applied to image processing problems.
Regular neural network has:
Input layer: accepts input
Hidden layer: performs calculations
Output layer: delivers
Each layer has neurons which connect to neurons in the previous layer.

Convolutional:
Data is treated as spatial. Instead of every neuron connection to previous neuron connection, it is connected close to it and all have the same weight. This leads to simplification in the network and network upholds the spatial aspect of the dataset.
This refers to filtering, simplification for better processing and understanding.

It is made of multiple layers:

- Convolutional layer
- Pooling layer: Rectified Unit layer
- ReLU
- Fully connected layer

ReLU: Acts as activation function ensuring non linearity as data moves through the network, without it data being fed into system would lose dimensionality
Fully connected layers: Allows to perform classification on dataset
Convolutional layer:

- Place a filter over an array of image pixels.
- This creates a convolved feature map.
- It is like looking at an image through a window to look at specific features.

Pooling layer:

- Reduces sample size of a particular feature map
- Makes processing much faster
- Reduces number of parameters network needs to process
- Output of this is a pooled feature map. There are two ways of doing this:
    - ➔ Max pooling: Takes maximum input of a particular convolved feature
    - ➔ Average pooling: Simple average

These steps amount to feature extraction where the network builds up a picture of the image data according to its own mathematical rules.

For performing classification, you need to move to the fully connected layer.

How to train CNN?
Unlabeled data: Unsupervised learning methods
Autoencoders: using these allows to squeeze data in a space with low dimensions
Performing calculations in the first part of the CNN
Reconstruct with additional layers that up-samples the data you have