Cascaded Deep Decision Networks for Classification of Endoscopic Images

Venkatesh N. Murthy[a], Vivek Singh[b], Shanhui Sun[b], Subhabrata Bhattacharya[b], Terrence Chen[b], and Dorin Comaniciu[b]

[a]School of Computer Science, University of Massachusetts, Amherst, USA.
[b]Medical Imaging Technologies, Siemens Healthcare, Princeton, USA

ABSTRACT

Both traditional and wireless capsule endoscopes can generate tens of thousands of images for each patient. It is desirable to have the majority of irrelevant images filtered out by automatic algorithms during an offline review process or to have automatic indication for highly suspicious areas during an online guidance. This also applies to the newly invented endomicroscopy, where online indication of tumor classification plays a significant role. Image classification is a standard pattern recognition problem and is well studied in the literature. However, performance on the challenging endoscopic images still has room for improvement. In this paper, we present a novel Cascaded Deep Decision Network (CDDN) to improve image classification performance over standard Deep neural network based methods. During the learning phase, CDDN automatically builds a network which discards samples that are classified with high confidence scores by a previously trained network and concentrates only on the challenging samples which would be handled by the subsequent expert shallow networks. We validate CDDN using two different types of endoscopic imaging, which includes a polyp classification dataset and a tumor classification dataset. From both datasets we show that CDDN can outperform other methods by about 10%. In addition, CDDN can also be applied to other image classification problems.

Keywords: Medical Image Classification, Colonoscopy, Confocal LASER Endoscopy, Deep Learning.

1. INTRODUCTION

Endoscopic image analysis continues to play a quintessential role in visual diagnosis of medical conditions originating primarily in the gastrointestinal, respiratory, or other vital tracts of the human body. Early and precise detection of a plethora of these conditions, can increase the chances of survival of an ailing patient through appropriate clinical procedures. For example, the relative 5-year survival rate for Colo-Rectal Cancer, when diagnosed at an early Polyp stage before it has spread, is about 90%.[1] Similarly, Meningioma, a benign intra-cranial tumor condition, occurring in approximately 7 of every $100,000$ people ,[2] if detected early, can be treated surgically or by radiation, thereby drastically reducing the chances of growth and potential transformation to malignancy.

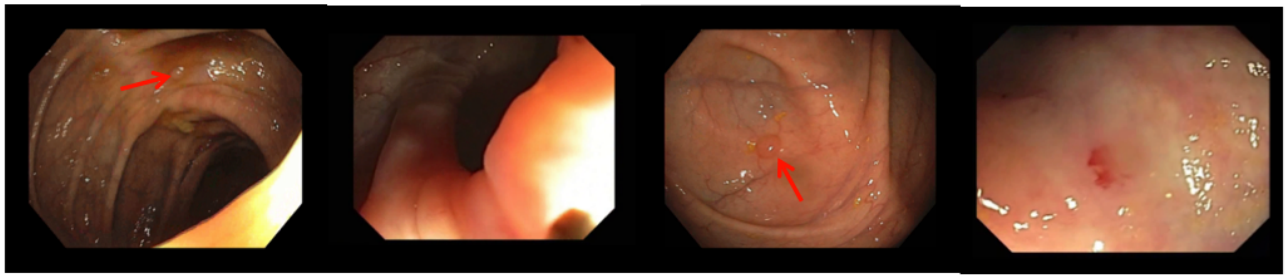


Figure 1. Sample images from the Polyp Classification Dataset obtained during a typical colonoscopic examination. Note the translucent blob-like shapes (pointed by arrows in red color) indicates a colon polyp.

Currently, clinicians visually scan endoscopic images, usually captured through electro-optical probes, for abnormal cell or tissue growth in the region under observation. Such manual screening procedures can often

Further author information: (Send correspondence to Venkatesh N. Murthy, E-mail: venk@cs.umass.edu)

become tedious as a single probe typically generates multitude of images. Furthermore, since the screening relies heavily on the dexterity of the clinician in charge, cases of miss detection are not uncommon. This emphasizes on an inevitable necessity of computer aided diagnostic (CAD) solutions that can not only efficiently minimize human effort required while screening a large fraction of negative cases, but also provide reliable reference to the clinicians. In this work, we focus only on eliminating negative images and all the experimental results are reported based on this.

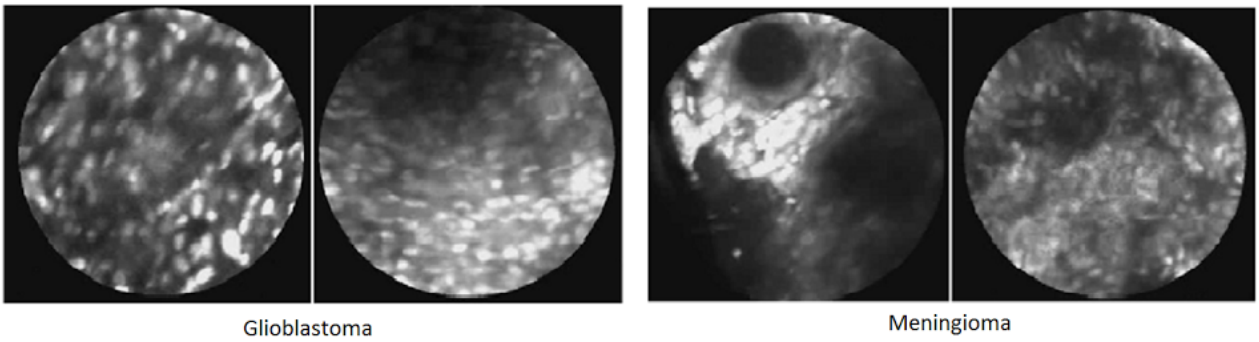


Figure 2. Sample Confocal LASER Endoscopic images from the Tumor Classification dataset with malignant Glioblastoma cases on the left and beningn Meningioma cases on the right. Note the sharp granular texture patterns in Glioblastoma cases.

In practice, each endoscopic procedure is specific to the medical condition and region of the body under observation. For example, within Capsule Endoscopy ,[3] an encapsulated wireless video camera is used to capture images from the gastrointestinal tract. In a different vein, neurosurgeons employ Confocal Laser Endomicroscopy (CLE)[4] probes as a surgical guidance tool to examine brain tissues for intracranial tumors. Although, these application scenarios are vastly different, their fundamental objective involves searching for visually discriminative patterns that can be decisive for a binary classification task primarily to segregate positive from negative image samples.

More specifically, we focus on the following two tasks: (1) In colonoscopic images, the objective is to filter out a large number of images that do not contain colon polyps (visually translucent blobs in the GI tract as seen in Fig. 1), and (2) Identify malignant cases of brain tumors (Glioblastoma, often identified by sharp granular patterns) from the benign ones (Meningioma, characterized by smooth homogeneous patterns) in CLE images containing either of the two (refer to Fig. 2). Both of these scenarios have their own challenges. The former case has several non-trivial inhibitors encountered by current computer vision systems: non-uniform illumination from light emitting diodes, noise from bubbles, bowel fluids, occlusion posed by anatomical complexity, large degrees of variation in shape and size. The latter is limited with the low resolution of current CLE imagery, motion artifacts and often presence of both kind of patterns in the probing area.

Automatic visual analysis of images pertaining to the aforementioned domains using conventional computer vision based techniques has demonstrated reasonable success in the past. Most of these are based on variants of Bag of visual Words (BoW) based computational frameworks owing to their simplicity of implementation. These methods[5–9] typically involve extraction of features from image, followed by a vector quantization step based on a pre-defined visual vocabulary (usually constructed by k-means clustering) which results in an intermediate compact representation of an image that can be ingested as a training sample for supervised classifiers. While these methods are effective, they consistently fail to leverage the data-driven aspect of the problem as all three steps - feature extraction, generation of intermediate representation, and finally the classification, are mutually independent.

Recently, Deep Learning based approaches,[10] have demonstrated significant performance boost on generic image classification tasks [11] by addressing the final classification objective in an integrated framework using layered neural networks. This has motivated a lot of researchers to apply deep neural network based methods in the field of medical image analysis.[12–17] In an early work [18] pertinent to classification, the authors introduce a

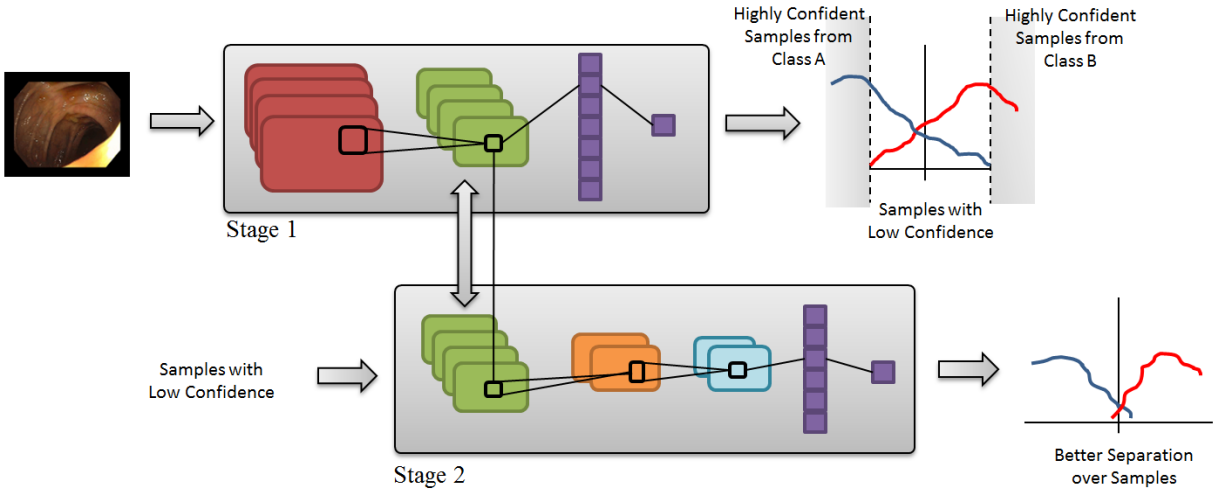


Figure 3. Cascaded Deep Decision Network (CDDN). For instance, stage-2 is built on top of conv layer (green color) of stage-1.

two-layer network which utilizes independent subspace analysis to reconstruct a natural representation of tumor images captured through cell-microscopy.

With that being said, training networks for medical image classification tasks is a challenging task as it often requires thorough experimentation on large datasets. Due to the lack of large amount of good quality training data, the trained network architecture often overtly optimizes itself for only training data, and performs poorly on unseen test samples. The authors of[19] avoid this issue by employing a pre-trained Convolutional Neural Network[10] whose parameters are learned from a large database of images from non-medical cases.[11] Their research demonstrates high performance on a medical application of chest pathology detection in X-ray images. We argue that while such a pre-trained architecture has demonstrated success in a specific cross-domain exercise, the generalization aspect is still inconclusive. In this paper, we propose a novel elegant computational framework called Cascaded Deep Decision Network (CDDN) to design an efficient network architecture with limited data yet without over-fitting characteristics during the training process. In contrast to existing deep learning based approaches, CDDN is built stage-wise during the learning phase. Our approach leverages a sampling strategy that discards samples classified with high confidence by a pre-trained network at the first-stage. Successive expert networks at different stages are trained focusing on samples that are difficult to classify. This work is inspired by decision trees[20] and boosting[21],[22] which are both classical approaches in machine learning. Many variants of boosting trees have been explored and are shown to be successful for most of the vision tasks[2122].[23] The fundamental concept of cascading is, early rejection of majority of test examples that has been widely utilized to achieve real-time performance. Hence, we provide an efficient and effective approach to utilize this concept in the context of deep learning. Specifically our contributions are as follows: (a) piece-wise training strategy helps alleviate problems encountered by gradient based methods, used heavily in contemporary deep learning research, (b) The proposed network architecture can make early decision thereby significantly reducing computational time without compromising on the performance, (c) data-driven design of CDDN offers an insight into underlying structure in the data and finally (d) we demonstrate the effectiveness of our approach through rigorous experiments on two extremely challenging endoscopic image classification tasks.

On the philosophical perspective, our proposed approach derives some similarity with ensemble methods commonly used in machine learning.[24,25] However, a majority of these approaches encounter difficulties rejecting outliers in presence of noisy training data. The sample selection strategy in CDDN facilitates circumventing this issue early on, thereby not affecting the final performance of the network. To the best of our knowledge, this is the first work that introduces flavors of cascading deep networks [26,27] into computer aided diagnosis of two crucial medical imaging applications. Extension of this framework to address multi-class classification problem is provided in one of the recent work.[28]

2. METHODOLOGY

Given a classification task, training a performant deep network is a difficult task since there are no well established guidelines to design the network architecture. Thus, training a network involves a thorough experimentation and statistical analysis. Although going deeper in the neural network design has shown to be effective[29] but also at the same time it increases the risk of over-fitting. Furthermore, as we experiment with the network architecture during the training process, it is difficult to leverage the results of the network trained in previous iteration. To this end, we propose an alternate learning strategy to learn a deep neural network which allows building on and taking advantage of previous training experiments.

2.1 Cascaded Deep Decision Network (CDDN)

A cascaded deep decision network is a multi-stage deep neural network, with decision stumps at each stage to classify easily separable data earlier in the network. Overview of the CDDN computational framework is provided in the Figure 3.

Given a dataset, stage-1 (root) network is trained using the back propagation algorithm. Instead of optimizing the network to obtain the best performance, we only need to optimize until a reasonable performance is achieved e.g. 60-70%. Alternatively, a pre-trained network can be used as a stage-1 network if it achieves reasonable performance. The samples classified with high confidence are no longer considered for subsequent training. Further, a stage-2 network is trained to correctly classify the previously misclassified samples and/or the samples classified with low confidence; note that stage-2 network is only optimized on a subset of the training data which was considered difficult by stage-1. This has the effect that as we go deeper we continue to "zoom-in" on resolving the problem cases. This stage-wise process is then continued until desired performance is achieved.

There are several key differences between the CDDN architecture and the traditional deep networks. For instance, as we go deeper, the newly introduced layers gets trained only on the subset of the data. All the layers in previous stages are frozen while training the current stage. Furthermore, each subsequent stage builds on the feature space trained in the previous stage. Note that subsequent stage can also be trained starting from any layer of the previous stage, which can be determined using a cross validation data set.

2.2 Piece-wise training for CDDN

The proposed architecture is trained in a unique fashion, starting with a root network which is trained in a traditional way, we use the softmax layer to compute its performance and learn a threshold of confidence score for classification using cross-validation. The cross validation at each stage of the network is setup as follows: in the first stage, the training data is split into training (90%) and validation (10%) set, while the network gets trained on the training set, the confidence score is determined using the validation set. For the next stage training, we mix up both training and validation set of the previous stage and create a new split to continue the training process. This way we make sure that the entire training dataset gets utilized for training and as well as the threshold value at each stage is determined based on the unseen samples which comes into effect during testing.

At each stage, the samples with a confidence value below a threshold value are considered to be as hard samples or confusion cases. These will be handled by the subsequent expert network which could be as simple as a single layer or a composition of multiple convolutional layer along with fully connected layers. In this work, we consider a shallow network as the expert network consisting of a convolutional and two fully connected layers along with some non-linearity and dropout layers. We continue to train the subsequent network layers using only the hard samples. While we do this, we completely freeze the previously trained layers. In other words, we set the learning rate of the previously trained network to zero and only train the newly added layers, this process can be recursively implemented until there are no more hard samples in the training dataset or until the desired depth of the network is met. This way, we are able to make use of the previous layers efforts and also have the benefit of making an early decision based on the confidence score (provided by the softmax layer). The proposed training helps in overcoming the over-fitting problem during the training of expert shallow networks which concentrates only on the subset of the entire dataset. In addition, it also helps in avoiding the gradient optimization getting stuck in poor solutions and most importantly it provides better generalization, which is validated by our experimental evaluations.

2.3 Classification using CDDN

Given an image, we feedforward it through the first stage of the CDDN and obtain the confidence score from the softmax layer, If the score is higher than the threshold value (determined during the training process) then we declare it as final output. If not, we continue onto the next stage in the network and repeat the process until the last layer to get the final response. Mathematically,

$$f(I) = \begin{cases} y & \text{if } (\hat{I}_{s_{j=1}} = f_{s_{j=1}}(I)) > T_{s_{j=1}}\{i\} \\ y & \text{if } (\hat{I}_{s_{j=2}} = f_{s_{j=2}}(\hat{I}_{s_{j=1}})) > T_{s_{j=2}}\{i\} \\ & \vdots \\ y & \text{if } (\hat{I}_{s_{j=n}} = f_{s_{j=n}}(\hat{I}_{s_{j=n-1}})) > T_{s_{j=n}}\{i\} \end{cases}$$

where the above mentioned parameters are defined as follows: I: input image, y: predicted label, s_j: different stages of the network and $j \in 1 \ldots n$, n: number of stages, $f(.)$: embedding function representing the network that predicts class labels with confidence, $\hat{I}$: embedded image and $T_{s_j}\{i\}$: threshold of a class label i at stage s_j.

2.4 Experimental Validation on MNIST digits

To validate CDDN and to provide more insight, we carried out a simple binary classification of digits '6' and '8' from MNIST dataset.[30] Training set consists of 11769 samples and testing set has 1932 images. Here we considered LeNet has our starting stage-1 network and for every subsequent stages we added a convolution layer and a fully connected layer (going deeper but to handle only subset of the data which are considered to be the hard ones). In Figure 4 we can see that at stage-1 network, 11522 samples in the training and 1884 samples in the testing were classified with high probability (i.e., easy samples) and the remaining samples of 247 training and 48 testing were considerably hard to discriminate. Now, we build an expert network stage-2, which is built upon stage-1 feature space. Since the resulting network is data-driven, the stopping criterion for network-growth is when the subsequent network fails to discriminate or there are very few training samples left out. The hard samples resulting from stage-1 and subsequent layers are shown in Figure 5. We can clearly see that the stage-1 had some confusion cases which were resolved by the subsequent stage-2. Hence, in addition to improving the classification, the proposed approach provides some insight into the distribution of the samples.

Table 1. Quantitative Performance Comparison on Tumor Classification Dataset; ImageNet pre-trained features were reported using 'Conv4' [10] layer with Linear SVM.

	SIFT+BOW +SVM(RBF)			ImageNet Pre-trained features			Traditional Deep Network			Deep Decision Network		
	Acc.	Sen.	Spec.	Acc.	Sen.	Spec.	Acc.	Sen.	Spec.	Acc.	Sen.	Spec.
split-1	81	0.96	0.71	67	0.90	0.50	78	0.91	0.69	81	0.87	0.76
split-2	63	0.97	0.49	61	0.94	0.47	66	0.93	0.69	73	0.97	0.63
split-3	82	0.91	0.75	89	0.97	0.86	77	0.77	0.77	89	0.90	0.88
split-4	98	0.98	0.97	95	0.96	0.94	93	0.93	0.93	97	0.95	1.0
split-5	77	0.70	0.84	83	0.73	0.92	74	0.79	0.69	85	0.70	0.99
Overall	**79**			**78**			**76**			**86**		

3. NETWORK ARCHITECTURE AND IMPLEMENTATION DETAILS

In this section, we provide all the required implementation details of our proposed method along with the baselines setup such as TDN, using ImageNet Pre-trained features with SVM and conventional approach of using BOW representation for SIFT with SVM.

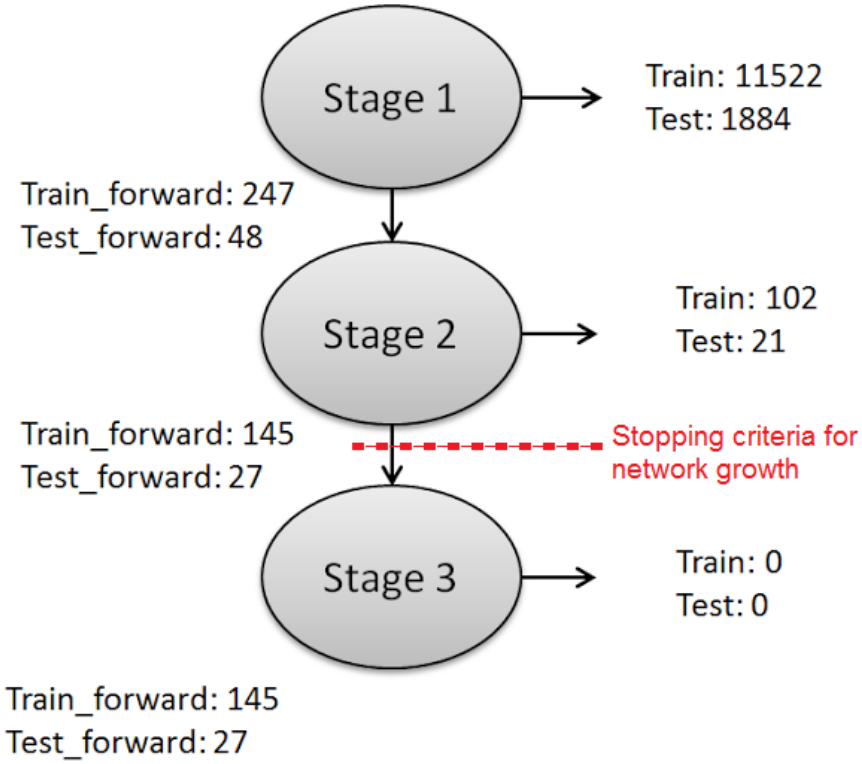


Figure 4. For validating the proposed method, CDDN was applied to the binary classification of digit '6' and '8' of MNIST dataset. One of the stopping criteria for network growth is when we see no improvement on the validation/training dataset performance, hence in this case it will result in two-staged network.

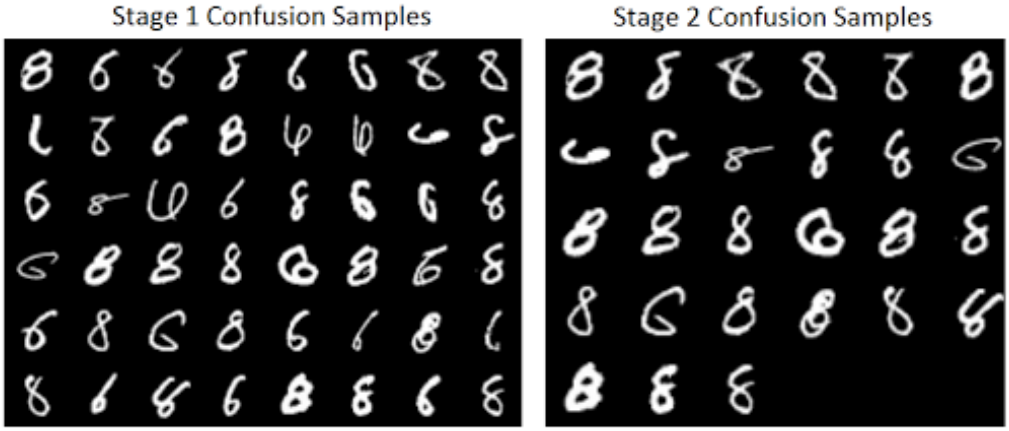


Figure 5. CDDN method idea validation on classification of digit '6' and '8' of MNIST dataset. left image indicates some of the confusion classes at stage-1 and the right one indicates some confusion cases at stage-2. One could observe that some of the confusion cases of stage-1 are resolved at stage-2.

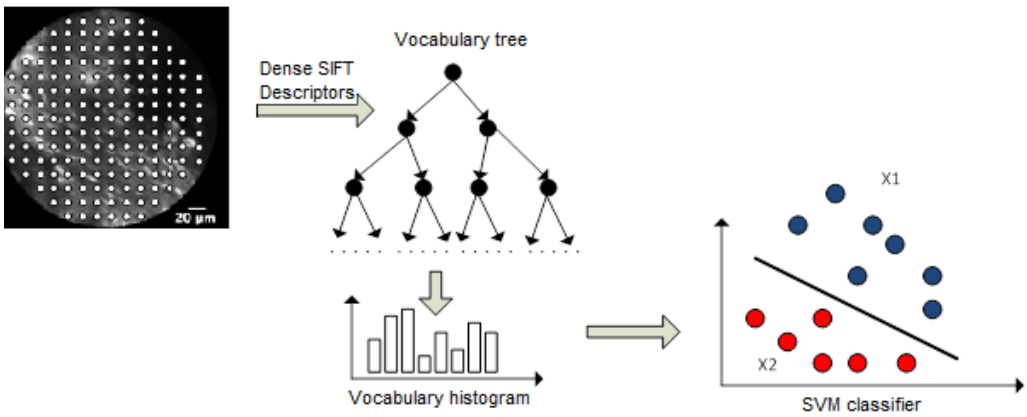


Figure 6. Workflow of BOW representation for DSIFT with SVM classifier. White dots in the image represent the sampling points.

3.1 Bag-of-Words SIFT feature with SVM

For a given image, Dense SIFT (DSIFT) descriptors of 128 dimension are computed for every n_s pixels inside the region of interest R of each image. where R is the lens area and n_s is the sub-sampled pixels. Further, a modified vocabulary tree structure[31] is utilized to construct a visual vocabulary dictionary. The vocabulary tree defines a hierarchical quantization using a hierarchical k-means clustering. In this work, a complete binary ($k = 2$) search tree structure is utilized. 2^{n_d} leaf nodes are finally used as visual vocabulary words, where, n_d is the depth of the binary tree. In the vocabulary tree learning stage, first the initial k-means algorithm is applied to the training data (a collection of SIFT descriptors derived from training data set. We randomly selected subset of the samples from these descriptors for final training) and then partitioned into 2 groups, where each group consists of SIFT descriptors closest to the cluster center. This process is then recursively applied until tree depth reach n_d. In the online stage, a SIFT descriptor (a vector) is passed down the tree by each level via comparing this feature vector to the 2 cluster centers and choosing the closest one. The visual word histogram is computed for all the dense SIFT descriptors on each image. The resultant quantized representation is used to train an SVM classifier with a RBF kernel. The parameters of the SVM classifier are chosen using a coarse grid search algorithm. The entire workflow is depicted in the Figure 6 for brain tumor classification data and we use similar kind of setup for the polyp classification as well.

3.2 ImageNet Pre-trained Features with SVM

For an image, we extract feature vectors from all the layers of a pre-trained CNN on ILSVRC-2012 dataset.[32] The dataset contains 1.2 million images which are manually annotated with labels from 1000 words vocabulary. Features are computed by forward propagating a mean-subtracted 224x224 RGB image through eight convolutional layers and three fully connected layers. In our case, we resize all the images irrespective of their aspect ratio to 224x224 to make it compatible with pre-trained CNN. Features extracted from various layers were fed to the linear SVM classifier to evaluate its classification performance. This study was conducted to evaluate the performance of off-the-shelf pre-trained CNN features when applied to a couple of medical image classification problems and this also serves as a baseline.

3.3 Traditional Deep Network (TDN) and Cascaded Deep Decision Network (CDDN)

We used different deep network architectures to solve polyp/no-polyp and meningioma/glioblastoma classification problems. The network architecture is summarized in Table 2. Notice that in the second stage, a convolution layer (Conv3) is introduced after the Conv2 layer, followed by fully connected (FC) layers. During stage 2 training, all the layers before Conv3 were frozen and the subsequent FC layers were randomly initialized. The final network architecture was determined based on performance on a validation dataset. For all experiments, the step learning rate policy was adopted with the following parameters: learning rate set to 0.001, step size of 10000 and momentum of 0.9. The training loss converged well for both the datasets.

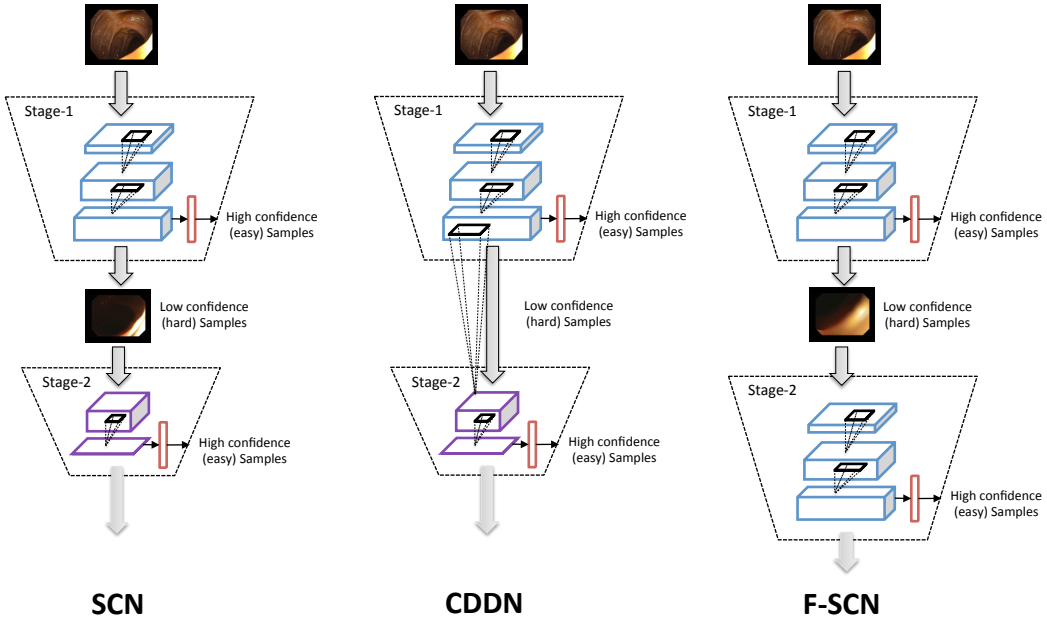


Figure 7. Comparison between Network Architectures. From left to right, Simple Cascaded Network (SCN), Cascaded Deep Decision Network (CDDN) and Fine-tuned Simple Cascaded Network (FSCN). Notice that CDDN's stage-2 is built on previous stage's feature space but for others cascade networks, the stage-2 is trained starting again from original image. For F-SCN, the first stage and second stage have the same architecture (depicted by color).

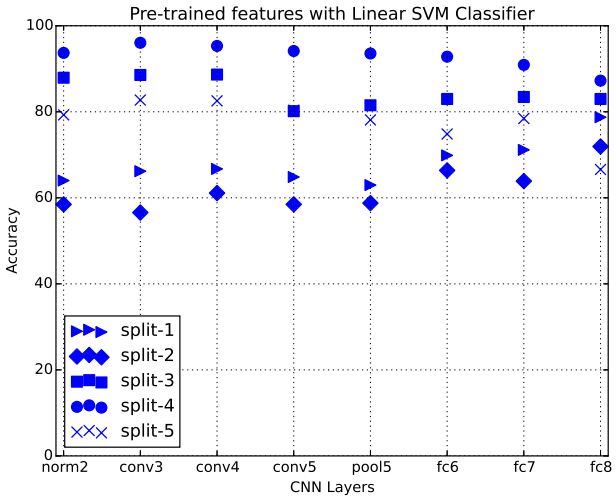
For comparison with traditional deep neural networks, we also train a deep network with similar model complexity as CDDN, in terms of number of layers and weight parameters. Thus, all the stage-1 and stage-2 layers of CDDN are combined to obtain a deep neural network, referred as TDN in our experiments. This network (TDN) serves as a strong baseline to CDDN, since starting from stage-1 deep network, TDN can be interpreted as a classic "going deeper" alternative to CDDN (which instead learns a stage 2 network on a subset of samples).

3.4 Cascaded Network

Cascading is a type of ensemble learning which involves concatenation of several classifiers. It is a multi-stage (classifier at each stage with same or different feature) approach, where the output information of the classifier is fed to the next classifier in the cascade. Since our approach bear similarities to cascading, we present alternate deep network architectures that directly embody cascading (ensemble of deep network classifiers), and provide a comparison with CDDN. We refer to these networks as simple cascaded networks (SCN) and fine-tuned simple cascaded network (F-SCN). Figure 7 provides a comparison between CDDN and simple cascaded network architectures (SCN and F-SCN).

Simple Cascaded Networks (SCN): This network is realized as a cascade of deep network classifiers, where each stage network is trained only on the misclassified samples from previous stage. Unlike CDDN where each stage builds on the feature space of the previous stage, SCN trains the network in every stage starting from the original image; hence the correlation between the networks across stages is weaker in SCN. To enable a direct comparison, the size of the network (number of parameters) at each stage of SCN and CDDN is kept same in all the experiments (ensuring similar model complexity).

Figure 8. Classification Accuracy of different layers of pre-trained network as features with SVM classifier for Brain Tumor Classification



Fine-tuned Simple Cascaded Networks (F-SCN): Similar to SCN, this network is also realized as a cascade of deep network classifiers. However, instead of using a shallow network in subsequent stages, F-SCN duplicate the previous stage's network (including the parameters) and fine-tune the parameters to correct the misclassified samples from previous stage. In other words, a 2 stage F-SCN has two deep CNN networks with similar architecture (for network details in each stage please refer to Table 2). Similar to SCN, F-SCN trains each stage starting from the original image. The motivation behind this cascaded network design with fine-tuned networks at each stage is to help avoid over-fitting/under-fitting, since the number of training samples reduced after each stage and deep networks are known to easily over-fit on smaller datasets. Notice that the 2 stage F-SCN has almost twice the number of parameters network compared to CDDN (since stage-1 are generally much larger than stage-2) resulting in an increased model complexity and computational time.

Table 2. CDDN Configuration details. Conv: Convolutional layer, FC: Fully connected layer, AvePool: Average pooling and MaxPool: Max pooling. Each Conv layer is followed by a nonlinear function ReLU. Except for the last FC layer, rest of the FC layers are followed by ReLU and dropout layer with p=0.5.

Dataset	Convnet Configuration									
Polyp	stage-1	image (92x110x3)	Conv1 (64x11x11)	Maxpool (3x3)	Conv2 (128x5x5)	Avepool (3x3)	FC (512)	FC (2)		
	stage-2						Conv3 (256x3x3)	AvePool (3x3)	FC (512)	FC (2)
Brain Tumor	stage-1	image (110x110x1)	Conv1 (96x11x11)	MaxPool (3x3)	Conv2 (256x5x5)	MaxPool (3x3)	FC (4096)	FC (4096)	FC (2)	
	stage-2					Conv3 (384x3x3)	FC (4096)	FC (4096)	FC (2)	

4. EXPERIMENTS

We report performance of our proposed method in comparison to other methods on two different setup for endoscopic imaging - Brain tumor classification (classify images into Meningioma or Glioblastoma) and Polyp classification (to flag images containing a polyp). In both cases, we report results using bag of visual words (BOW) SIFT feature with SVM (RBF kernel and ImageNet pre-trained features (Best performing layer) with SVM. In addition, we report results using our proposed method CDDN and the strong baseline TDN (all the stages/network layers combined). Please note that, in order to have a fair comparison, both the TDN and CDDN was designed to have the same complexity (number of layers and parameters).

4.1 Tumor Classification

Dataset: We use a commercially available clinical endo-microscope in the market called Cellvizio (Mauna Kea Technologies, Paris, France). Cellvizio is a probe-based CLE system. It consists of a laser scanning unit, proprietary software, a flat-panel display and fiber optic probes providing a circular field of view with a diameter of 160μm. The device is intended for imaging the internal micro-structure of tissues in the anatomical tract that are accessed by an endoscope. The system is clinically used during an endoscopic procedure for analysis of sub-surface structures of suspicious lesion, which is primarily referred to as optical biopsy [20]. In a surgical resection application, a neurosurgeon inserts a hand-held proof into a surgical bed to examine the remainder of the tumor tissue to be resected.

The equipment is used to collect 117 short videos, each from a unique patient suffering from Glioblastoma and relatively longer videos from patients with Meningioma. All videos are captured at 24 frames per second, under a resolution of 464x336. The collection of videos are hereafter being referred to as the Brain Tumor Dataset.

Pre-processing: Due to the limited imaging capability of CLE devices or intrinsic properties of brain tumor tissues, the resultant images often contain little categorical information and are not useful for recognition algorithms. Image entropy has been constantly used in the past[33] to quantitatively determine the information content of an image. Specifically, low-entropy images have very little contrast and large runs of pixels with the same or similar values.
In order to filter uninformative video frames, we empirically determine an entropy threshold, by calculating the distribution of the individual frame entropy throughout the dataset (calculated over 34, 443 frames). In our case, this threshold is 4.15. This simple thresholding scheme allows us to select 14,051 frames containing Glioblastoma and 11,987 frames containing Meningioma cases. Experimental results are provided based on a leaving a pair of patients (one with Glioblastoma and other with Meningioma) out. Further, we took a center crop of 220x220 square image inscribed in the circular lens region. Please note that for all the deep learning related experiments, images were resized to 110x110x1 to reduce the computational complexity.

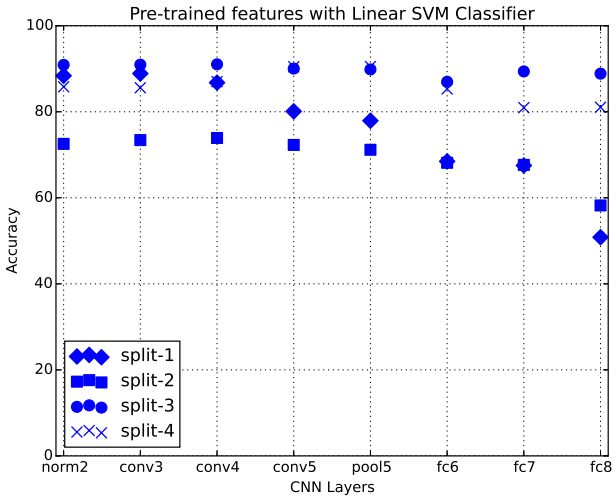
Discussion: See table 1 for performance comparison. It is clearly evident that our proposed method CDDN significantly outperforms all the other methods. In comparison to TDN, CDDN improves the performance by around 9%, it does well on all the three measures of accuracy, sensitivity and specificity. This provides the evidence that our proposed method of building deeper networks is better than the traditional way of going deeper. Since CDDN makes early decision on several samples, the average processing time for each sample for CDDN is lower as compared to TDN. We also provide the evaluation of different layers of the Pre-trained network as features with SVM classifier in the Figure 8. We can see that on an average 'Conv4' layer performs better across all the splits and hence to be consistent we report their results in the Table 1 as a baseline.

4.2 Polyp Classification for Colonoscopy

Dataset: Results are reported on a publicly available Polyp dataset from ISBI 2014 Challenge on Automatic Polyp Detection in Colonoscopy Videos.[34] The dataset consists of 21 short colonoscopy videos from ASU-Mayo Clinic polyp database, of which 11 videos have a unique polyp inside (positive shots) and the other 10 videos have no polyps (negative shots). Some videos are high resolutions but some are recorded in lower resolution, some videos display a careful colon examination while others show a hasty colon inspection, finally some videos have biopsy instruments in them. Please note that, even the videos containing polyp will have a large number of frames where polyp is absent and hence groundtruth labels are provided at frame level. In our evaluation, we provide experimental results on four random splits (are at video level to avoid bias during train and test split) by reporting classification accuracy at frame level and also provide ROC curves.

Pre-processing: Since the videos were of different resolutions and region around the frames were varying, we fixed the final image size to be 636x530 (chosen based on the average resolutions of all the video frames. We identified the lens region separated from rest of the black region and then resized (maintaining the aspect ratio) to fit the fixed window size of 636x530. Since frames containing polyp were relatively very low we chose to perturb only the positive (contains polyp) frames. Perturbation involved rotation by angles of 90,180 and 270 degrees followed by flip and again rotate with the same set of angles. Please note that for all the experimentation the resulting image were later resized to 110x92x3 to handle the computational complexity.

Figure 9. Classification Accuracy of different layers of pre-trained network as features with SVM classifier for Polyp/No-Polyp Classification



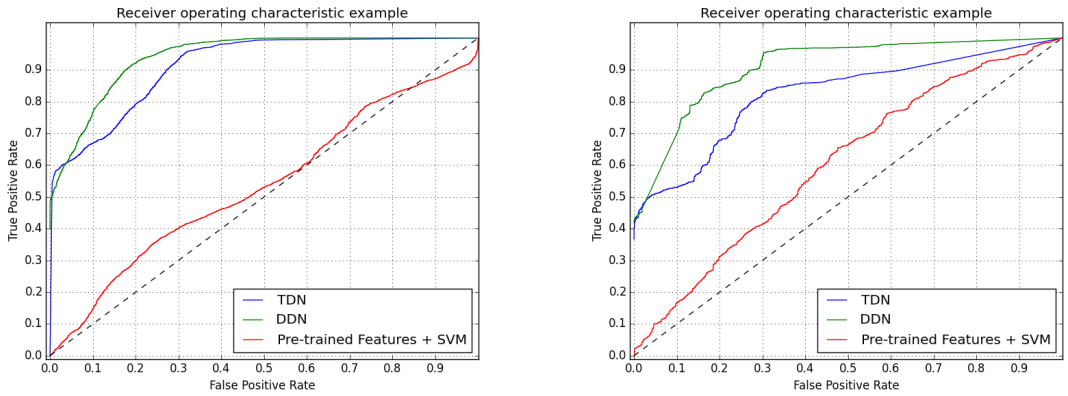
Discussion: Table 3 demonstrates the performance comparison. We observe similar performance trends as reported for brain tumor classification, where our proposed method CDDN outperform all the other methods. In addition to accuracy metric, we have also provided the ROC curve for all the splits in Figure 10. Overall, area under the curve is significantly better for CDDN when compared to rest of the methods. All these experimental results convey that the proposed CDDN method is an efficient and effective alternative to traditional way of building a deeper network. Considering a clinical use case, if we pick an operating point of false positive rate=17% with true positive rate=90%, then our system on an average is able to eliminate 84% of the negative images (do not contain polyp) but still be able to identify 90% of the positive cases (containing polyp) accurately. In Figure 9 we provide the effectiveness of different layers of the Pre-trained network as features when combined with SVM classifier. On an average across all the splits, we found that 'Conv3' layer gives the best performance and thus their results are reported in the Table 3 as a baseline.

Table 3. Quantitative Performance Comparison on Polyp Classification Dataset

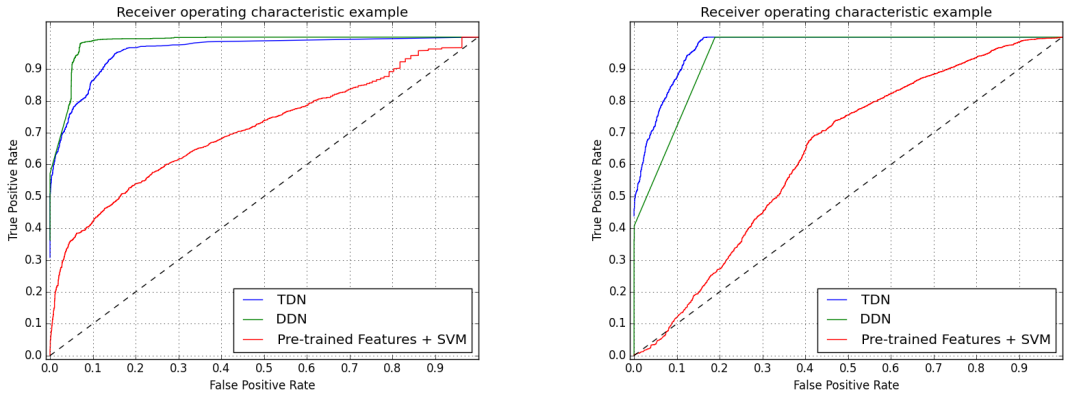
	SIFT+BOW +SVM(RBF)	ImageNet Pre-trained features (Conv3)	TDN	CDDN
	Acc.	Acc.	Acc.	Acc.
split-1	89.1	88.89	78.34	87
split-2	37.46	73.41	67.81	83
split-3	70.82	90.95	88.88	92.75
split-4	82.90	85.59	84.45	92.40
Overall	**70.08**	**81.66**	**80.67**	**87.43**

4.3 Comparison with SCN and F-SCN

The stage-wise performance comparisons of our proposed method CDDN in comparison to SCN and F-SCN are provided in Table 4 and Table 5 for polyp and tumor classification dataset respectively. We can observe that CDDN outperforms both SCN and F-SCN at each stage for all splits and its better even in terms of overall performance. We believe that SCN and F-SCN couldn't perform well because of over-fitting/under-fitting problem in the second stage due to limited number of samples (hard samples). This clearly indicates that our proposed method has the ability to dodge this prevalent problem while applying deep learning networks for medical related



(a) Split-1 (b) Split-2



(c) Split-3 (d) Split-4

Figure 10. ROC curves for Polyp dataset across all the splits.

problems where the data is limited.

Table 4. CDDN Performance analysis on Polyp Classification Dataset. # - number of samples; Acc. - Accuracy (%).

	SCN					F-SCN					CDDN				
	Stage-1		Stage-2		Overall	Stage-1		Stage-2		Overall	Stage-1		Stage-2		Overall
	#	Acc.	#	Acc.	Acc.	#	Acc.	#	Acc.	Acc.	#	Acc.	#	Acc.	Acc.
split-1	3373	96.02	4090	67.41	67.41	4696	97.84	2767	58.07	83.1	3712	98.94	3751	86.36	83.36
split-2	1248	92.62	1595	51.28	51.28	1227	92.25	1616	50.99	68.8	1501	94.53	1342	83.07	83.07
split-3	1740	100	3665	62.49	62.49	3830	98.09	1575	63.23	87.93	4245	99.74	1160	92.74	92.74
split-4	3325	99.06	2629	66.67	66.67	3593	98.71	2361	75.91	89.46	3391	99.97	2563	92.40	92.40

4.4 Computational efficiency of CDDN during testing phase

Let us assume, Q input feature maps produces R output feature maps, and the feature map size is M M (being equal for simplicity). Let the convolution kernel size be KxK. If we consider CDDN, then in the first stage there are 2x$\mathcal{O}(R \times Q \times M^2 \times K^2)$ computations for two convolutional layer and $2\mathcal{O}(Q \times M^2)$ for two pooling layers. In the second stage, there would be 1x$\mathcal{O}(R \times Q \times M^2 \times K^2)$ computation for one convolutional layer and 1x$\mathcal{O}(Q \times M^2)$ for one pooling layer. Since Fully connected layer computations remain same in the first and second stage we ignore that. Now consider TDN, since it consists of first and second stage put together (end-to-end learning) it involves 3x$\mathcal{O}(R \times Q \times M^2 \times K^2)$ and 3x$\mathcal{O}(Q \times M^2)$ computations.

Table 5. CDDN Performance analysis on Tumor Classification Dataset. # -number of samples; Acc. -Accuracy (%).

	SCN					F-SCN					CDDN				
	Stage-1		Stage-2		Overall	Stage-1		Stage-2		Overall	Stage-1		Stage-2		Overall
	#	Acc.	#	Acc.	Acc.	#	Acc.	#	Acc.	Acc.	#	Acc.	#	Acc.	Acc.
split-1	540	96.14	791	25.53	54.99	333	92.79	998	69.13	75.05	340	77.35	991	82.44	81.13
split-2	203	91.03	481	16.83	39.47	255	92.94	429	40.79	60.23	121	100	563	67.49	73.24
split-3	1384	86.56	1962	46.48	63.06	444	88.51	649	62.71	73.19	445	97.75	648	82.87	88.92
split-4	445	100	237	67.93	88.85	544	100	138	78.26	95.6	537	100	145	87.58	97.35
split-5	1367	87.63	1979	47.85	64.1	1507	90.31	1839	60.95	74.17	1177	99.15	2169	80.26	86.90

For instance, if we consider split-1 of Polyp classification classification using CDDN (see Table 4, we can observe that among all test samples, 3712 samples was able to advantage of early decision and rest of 3751 samples (hard cases) went to the second stage. So effectively, for 3712 samples we saved additional computations made in the second stage ($\mathcal{O}(R \times Q \times M^2 \times K^2)$and $\mathcal{O}(Q \times M^2)$) when compared to TDN. In case of TDN, for each of the test sample (irrespective of hard/easy case) it requires same number of computations (3x$\mathcal{O}(R \times Q \times M^2 \times K^2)$and 3x$\mathcal{O}(Q \times M^2)$). Hence, our proposed method is efficient making it more suitable for real-world applications.

5. CONCLUSION

We presented an efficient and effective alternative for building a deep neural network called CDDN. CDDN is built stage-wise by greedily discarding the samples classified with high confidence and only focusing on the confusing cases in the subsequent expert networks built at different stages. As an idea validation, we presented detailed experimental results on binary classification of digit '6' and '8' as part of MNIST dataset. Further, the proposed method was shown to outperform all the other methods on real world challenging problems such as polyp/no-polyp and meningioma/glioblastoma classification. One could also benefit from making early decisions in the deep network to meet the real-time performance with little compromise on the performance. The proposed approach can be in general applied to any image classification task.

REFERENCES

[1] Baxter, N. N., Goldwasser, M. A., Paszat, L. F., Saskin, R., Urbach, D. R., and Rabeneck, L., "Association of colonoscopy and death from colorectal cancer," *Annals of Internal Medicine* **150**(1), 1–8 (2009).

[2] Lee, J. H. and Sade, B., "Meningiomas of the central neuraxis: Unique tumors," in [*Meningiomas*], 157–162 (2009).

[3] Mishkin, D. S., Chuttani, R., Croffie, J., DiSario, J., Liu, J., Shah, R., Somogyi, L., Tierney, W., Wong Kee Song, L. M., and Petersen, B. T., "Asge technology status evaluation report: wireless capsule endoscopy," *Gastrointestinal Endoscopy* **63**(1), 539–545 (2006).

[4] Paull, P. E., Hyatt, B. J., Wassef, W., and Fischer, A. H., "Confocal laser endomicroscopy: a primer for pathologists," *Arch. Pathol. Lab. Med.* **135**, 1343–1348 (Oct 2011).

[5] Li, B., Meng, M.-H., and Xu, L., "A comparative study of shape features for polyp detection in wireless capsule endoscopy images," in [*Proc. of IEEE Eng Med Biol Soc*], 3731–3734 (2009).

[6] Andr, B., Vercauteren, T., Perchant, A., and Buchner, A. M., "Introducing space and time in local feature-based endomicroscopic image retrieval," in [*Medical Content-Based Retrieval for Clinical Decision Support*], *Lecture Notes in Computer Science* **5853**, 18–30 (2010).

[7] Andre, B., Vercauteren, T., Buchner, A. M., Wallace, M. B., and Ayache, N., "A smart atlas for endomicroscopy using automated video retrieval," *Med Image Anal* **15**, 460–476 (Aug 2011).

[8] Zhao, Q. and Meng, M.-H., "Polyp detection in wireless capsule endoscopy images using novel color texture features," in [*Intelligent Control and Automation (WCICA), 2011 9th World Congress on*], 948–952 (2011).

[9] Li, B. and Meng, M. Q.-H., "Automatic polyp detection for wireless capsule endoscopy images," *Expert Systems with Applications* **39**(12), 10952–10958 (2012).

[10] Krizhevsky, A., Sutskever, I., and Hinton, G. E., "Imagenet classification with deep convolutional neural networks," in [*Advances in Neural Information Processing Systems 25*], Pereira, F., Burges, C., Bottou, L., and Weinberger, K., eds., 1097–1105, Curran Associates, Inc. (2012).

[11] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L., "ImageNet: A Large-Scale Hierarchical Image Database," in [*IEEE CVPR*], (2009).

[12] Carneiro, G., Nascimento, J. C., and Freitas, A., "Robust left ventricle segmentation from ultrasound data using deep neural networks and efficient search methods," in [*Proceedings of the 2010 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, Rotterdam, The Netherlands, 14-17 April, 2010*], 1085–1088 (2010).

[13] Carneiro, G. and Nascimento, J. C., "Multiple dynamic models for tracking the left ventricle of the heart from ultrasound data using particle filters and deep learning architectures," in [*IEEE CVPR*], 2815–2822 (2010).

[14] Ngo, T. A. and Carneiro, G., "Left ventricle segmentation from cardiac MRI combining level set methods with deep belief networks," in [*IEEE International Conference on Image Processing, ICIP 2013, Melbourne, Australia, September 15-18, 2013*], 695–699 (2013).

[15] Carneiro, G., Nascimento, J. C., and Bradley, A. P., "Unregistered multiview mammogram analysis with pre-trained deep learning models," in [*MICCAI*], 652–660 (2015).

[16] Ghesu, F. C., Georgescu, B., Zheng, Y., Hornegger, J., and Comaniciu, D., "Marginal space deep learning: Efficient architecture for detection in volumetric image data," in [*Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015 - 18th International Conference Munich, Germany, October 5-9, 2015, Proceedings, Part I*], 710–718 (2015).

[17] Zheng, Y., Liu, D., Georgescu, B., Nguyen, H., and Comaniciu, D., "3d deep learning for efficient and robust landmark detection in volumetric data," in [*Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015 - 18th International Conference Munich, Germany, October 5-9, 2015, Proceedings, Part I*], 565–572 (2015).

[18] Le, Q. V., Han, J., Gray, J. W., Spellman, P. T., Borowsky, A., and Parvin, B., "Learning invariant features of tumor signatures," in [*9th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, ISBI 2012, May 2-5, 2012, Barcelona, Spain, Proceedings*], 302–305 (2012).

[19] Bar, Y., Diamant, I., Wolf, L., Lieberman, S., Konen, E., and Greenspan, H., "Chest pathology detection using deep learning with non-medical training," in [*12th IEEE International Symposium on Biomedical Imaging*], 294–297 (2015).

[20] Quinlan, J. R., "Simplifying decision trees," *International journal of man-machine studies* **27**(3), 221–234 (1987).

[21] Viola, P. and Jones, M., "Rapid object detection using a boosted cascade of simple features," in [*Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*], **1**, I–511, IEEE (2001).

[22] Schapire, R. E., "The boosting approach to machine learning: An overview," in [*Nonlinear estimation and classification*], 149–171, Springer (2003).

[23] Viola, P., Jones, M. J., and Snow, D., "Detecting pedestrians using patterns of motion and appearance," in [*Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*], 734–741, IEEE (2003).

[24] Freund, Y. and Schapire, R. E., "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.* **55**, 119–139 (Aug. 1997).

[25] Friedman, J. H., "Greedy function approximation: A gradient boosting machine.," *Ann. Statist.* **29**, 1189–1232 (10 2001).

[26] Sun, Y., Wang, X., and Tang, X., "Deep convolutional network cascade for facial point detection," in [*IEEE CVPR*], 3476–3483 (2013).

[27] Cui, Z., Chang, H., Shan, S., Zhong, B., and Chen, X., "Deep network cascade for image super-resolution," in [*Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*], 49–64 (2014).

[28] Murthy, V. N., Singh, V., Chen, T., Manmatha, R., and Comaniciu, D., "Deep decision network for multi-class image classification," in [*The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], (June 2016).

[29] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A., "Going deeper with convolutions," *CoRR* **abs/1409.4842** (2014).

[30] LeCun, Y. and Cortes, C., "MNIST handwritten digit database." http://yann.lecun.com/exdb/mnist/ (2010).

[31] Nister, D. and Stewenius, H., "Scalable recognition with a vocabulary tree," in [*Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*], **2**, 2161–2168, IEEE (2006).

[32] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T., "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093* (2014).

[33] Gonzalez, R. C. and Woods, R. E., [*Digital Image Processing*], Prentice-Hall, Inc., Upper Saddle River, NJ, USA (2008).

[34] Tajbakhsh, N., Liang, J., del Noza, J. B., and Gurudu, S. R., "Automatic polyp detection challenge in colonoscopy video. international symposium on biomedical imaging," (2015).