Digital Object Identifier 10.1109/ACCESS.2017.DOI

Confident Classification using a Hybrid between Deterministic and Probabilistic Convolutional Neural Networks

MUHAMMAD NASEER BAJWA[1,2,*], SULEMAN KHURRAM[1,2,*], MOHSIN MUNIR[1,2,*], SHOAIB AHMED SIDDIQUI[1,2], MUHAMMAD IMRAN MALIK[3,4], ANDREAS DENGEL[1,2],AND SHERAZ AHMED.[2]

[1]Technische Universität Kaiserslautern, 67663 Kaiserslautern, Germany
[2]German Research Center for Artificial Intelligence GmbH (DFKI), 67663 Kaiserslautern, Germany
[3]School of Electrical Engineering and Computer Science, National University of Science and Technology (NUST), 46000 Islamabad, Pakistan
[4]Deep Learning Laboratory, National Center of Artificial Intelligence, 46000 Islamabad, Pakistan
[*]Authors contributed equally

Corresponding author: Muhammad Naseer Bajwa (e-mail: naseer.bajwa@dfki.de).

This work is partially funded by National University of Science and Technology (NUST), Pakistan through Prime Minister's Programme for Development of PhDs in Science and Technology, BMBF project DeFuseNN (01IW17002), and NVIDIA AI Lab (NVAIL) programme.

ABSTRACT Traditional neural networks trained using point-based maximum likelihood estimation are deterministic models and have exhibited near-human performance in many image classification tasks. However, their insistence on representing network parameters with point-estimates renders them incapable of capturing all possible combinations of the weights; consequently, resulting in a biased predictor towards their initialisation. Most importantly, these deterministic networks are inherently unable to provide any uncertainty estimate for their prediction which is highly sought after in many critical application areas. On the other hand, Bayesian neural networks place a probability distribution on network weights and give a built-in regularisation effect making these models able to learn well from small datasets without overfitting. These networks provide a way of generating posterior distribution which can be used for model's uncertainty estimation. However, Bayesian estimation is computationally very expensive since it greatly widens the parameter space. This paper proposes a hybrid convolutional neural network which combines high accuracy of deterministic models with posterior distribution approximation of Bayesian neural networks. This hybrid architecture is validated on 13 publicly available benchmark classification datasets from a wide range of domains and different modalities like natural scene images, medical images, and time-series. Our results show that the proposed hybrid approach performs better than both deterministic and Bayesian methods in terms of classification accuracy and also provides an estimate of uncertainty for every prediction. We further employ this uncertainty to filter out unconfident predictions and achieve significant additional gain in accuracy for the remaining predictions.

INDEX TERMS Bayesian Estimation, Convolutional Neural Networks, Hybrid Neural Networks, Image Classification, Time-series Classification, Uncertainty Estimation

I. INTRODUCTION

OVER the last decade, Convolutional Neural Networks (CNNs) have made phenomenal strides in various classification tasks using a wide array of input modalities. These powerful algorithms have achieved impressive performance, often at par with human experts, in many challenging natural scene image recognition tasks [1]–[3] and even in sensitive and critical application areas like medical image analysis for disease prediction [4]–[8]. These CNNs gained significant attention due to their parameter efficiency, in contrast to other deep learning models like densely connected Multi-Layer Perceptrons (MLPs), resulting in comparatively better generalisation performance. They are particularly powerful in analysing visual modalities like images and videos [9] but have also proved their worth in time-series analysis where they have been used for classification [10] and anomaly detection [11].

The fundamental principle behind conventional CNNs is

to learn the optimal combination of network parameters (weights and biases) that can capture encoded representation of input training data. These conventional CNNs use point-estimates to represent network parameters and although they work astonishingly well in most image recognition tasks, they have large insatiable appetite for data [12]. Additionally, the $softmax$ function tips the odds in favour of one class by squashing classification probabilities for others. Therefore, it results in overly confident predictions often times even when the network is completely wrong. This compulsive behaviour of traditional point-based neural networks to always be relentlessly assertive in their prediction raises serious concerns in many crucial application areas like medical image analysis, security, autonomous driving, financial transactions and IoT (Internet of Things) based human health monitoring. Also, the very nature of these point-based classifiers prohibits them to associate uncertainty with their predictions, which is a highly desired characteristic of any AI-based classifier.

Bayesian estimation introduces a probabilistic perspective to the neural networks and addresses many shortcomings of traditional point-based neural networks. It represents each parameter with a probability distribution instead of a single point-estimate. As a result, Bayesian neural networks are able to learn effectively from relatively small amount of data and thus are fairly robust to overfitting [13]. They can provide an inherent regularisation effect [14] by constraining the network parameters within a distribution instead of letting them increase out of bound. Most importantly, Bayesian inference can allow to estimate network's uncertainty about any prediction. However, a full Bayesian estimation over all network parameters is computationally expensive and finding true posterior probability is intractable. These limitations are normally addressed by employing various tricks like Markov Chain Monte Carlo (MCMC) sampling [15] and Variational Inference (VI) [16], or a combination of the two [17], to approximate the true posterior with a manageable distribution. A CNN trained using Bayesian estimates for network parameters is shown to lag its counterpart, trained using point-estimates, in terms of classification accuracy [13], [18].

In this paper, we recognise specific merits of each approach discussed above and combine them into a hybrid training paradigm. This hybrid approach integrates deterministic CNNs, where each parameter assumes only one value, with probability driven Bayesian CNNs, where each parameter may take any value drawn from a probability distribution characterised by a mean and a standard deviation. This probability distribution is learnt for each parameter during training. The proposed hybrid training method provides an estimate of uncertainty, using Bayesian classifier, without compromising on classification accuracy owing to deterministic feature extractor. It also captures maximum weight configurations from small datasets while still remaining computationally manageable. The proposed approach is tested on 13 different classification datasets including benchmark image datasets, fine-grained medical image datasets and time-series datasets. The proposed hybrid method is proved to be superior to both fully deterministic and fully Bayesian CNN approaches in terms of classification accuracy.

A. RELATED WORK

Conventional CNNs have demonstrated their worth in various image recognition tasks since long [19] and have resurged into popularity in 2012 with Alexnet [20]. They have lately evolved into awfully complicated networks spanning thousands of layers [21].

Although applications of Bayesian method into neural networks have also been investigated for many decades [22], it was only after Blundell et al. [23] proposed Bayes by Backprop that training of deep neural networks was made possible using Bayesian estimation. This method of Variational Inference allowed backpropagation of so called Expected Lower BOund (ELBO) loss and regularising weight distributions. A CNN trained using Bayesian method was recently proposed by Shridhar et. al [18] as a fundamental construct for other network architectures. They used Bayes by Backprop for training convolutional network and reported comparable results on many benchmark datasets.

Acknowledging the excessive computational cost of Bayesian models, Gal and Ghahramani [24] proposed a Monte Carlo dropout method to approximate Bayesian inference in deep Gaussian processes. The method is equivalent to performing multiple forward passes through the network and taking the average of results to model the uncertainty of the network. Kwon et al. [25] recognised the importance of uncertainty quantification especially in medical domain and proposed to calculate it by splitting the uncertainty into aleatoric, which corresponds to model's uncertainty; and epistemic uncertainty, which represents inherent noise in the data. Kendall and Gal [26] studied the advantages of modelling epistemic uncertainty as compared to aleatoric uncertainty in deep Bayesian models.

Combining deterministic and probabilistic models in various fashions has also been studied for long. Tang and Salakhutdinov [27] pointed out that the conditional distribution $p(Y|X)$ does not need to be unimodel, as normally assumed by MLPs, but can also be represented as a multimodel output distribution for many structured prediction problems. They proposed a hybrid Sigmoid Belief Network (SBN) with some stochastic hidden variables and some deterministic hidden variables and achieved superior performance on synthetic and facial expression datasets. Similarly, other neural networks with partially Bayesian parameters have been proposed for regression tasks as alternative to Gaussian Processes [14], [28], which do not scale well with the number of training samples.

The problem of estimating uncertainty has been addressed in variety of ways, for example out-of-distribution (OOD) samples detection [29], [30] and density estimation using flow based models. Normalising flows and autoregressive models have been successfully combined to produce state-

of-the-art results in density estimation, via Masked Autoregressive Flows (MAF) [31]; and to accelerate state-of-the-art WaveNet-based speech synthesis to 20x faster than real-time [32], via Inverse Autoregressive Flows (IAF) [33]. Huang et al. [34] presented Neural Autoregressive Flows (NAFs) and demonstrated that these models are universal approximators for continuous probability distributions, and their greater expressivity allows them to better capture multimodal target distributions. Adding on to their work, Cao et al. [35] proposed Block Neural Autoregressive Flow which is a much more compact universal approximator of density functions, where a bijection is directly modelled using a single feedforward network. Dinh et al. [36] introduced a set of transformations called real-valued Non-Volume Preserving (real NVP) as a tractable and expressive way to modelling high-dimensional data. Ardizzone et al. [37] extended real NVP architecture and argued that their proposed Invertible Neural Networks (INNs) are well suited for determining full posterior parameter distribution conditioned on training data. They noted that alternating backward and forward training passes and accumulating gradients from both sides before updating parameters allows efficient training. Kingma et al. [38] furthered flow-based generative models [39] which are useful for calculating exact log-likelihood, performing exact latent-variable inference, and parallelising training and synthesis pipelines. Their Generative flow (Glow) model uses an invertible 1×1 convolution and is shown to be capable of efficient and accurate synthesis of large images.

II. PROPOSED HYBRID NEURAL NETWORK

A CNN primarily consists of two main modules: a feature extractor and a classifier. The proposed network consists of a set of convolutional layers trained with point estimates followed by fully-connected layers trained using Bayesian estimate. It provides a trade-off between high accuracy of deterministic models and uncertainty estimation of Bayesian models. It also restricts the parameter space of the network as compared to fully Bayesian models because only the classifier part of the network treats its parameters as random variables. Fig. 1 shows schematic diagram of the hybrid model proposed in this work. The network initially trains to optimise parameters for both convolutional feature extractor and dense classifier as given below.

$$\mathcal{W}_C^*, \mathcal{W}_D^* = \arg\min \mathcal{W}_C, \mathcal{W}_D \frac{1}{|\mathcal{X}|} \sum_{(\mathbf{x},y)\in\mathcal{X}\times\mathcal{Y}} \mathcal{L}\Big(\psi\big(\Phi(\mathbf{x};\mathcal{W}_C);\mathcal{W}_D\big), y\Big), \tag{1}$$

where $\mathcal{L}$ denotes the loss function, Φ represents the convolutional part of the network parameterised by $\mathcal{W}_C$ and ψ represents the dense layers (forming the classifier) parameterised by $\mathcal{W}_D$.

Once the network is trained using point-estimates, we reinitialise fully connected layers with random variables following normal distribution and retrain them using Bayesian estimation. The parameters of convolutional feature extractor are frozen throughout this retraining. This whole training paradigm allows us to capitalise on economically learned features by deterministic convolutional block and use expensive Bayesian inference only to approximate posterior distribution, which might then be used for uncertainty estimation. Mathematically, the learning of FC classifier of hybrid model is given by;

$$\theta_D^* = \arg\min \theta_D \frac{1}{|\mathcal{X}|} \sum_{(\mathbf{x},y)\in\mathcal{X}\times\mathcal{Y}} \mathcal{L}\Big(\Psi\big(\Phi(\mathbf{x};\mathcal{W}_C^*);\theta_D\big), y\Big), \tag{2}$$

where Ψ represents the Bayesian layers learned through Bayes by Backprop and θ_D denotes the distribution of weights. Since the weights are described by a distribution instead of point-wise estimates, $\mathcal{L}$ in this case denotes the ELBO loss. Convolutional feature extractor trained with point-estimates learns crisp features of the input data while probabilistic classifier allows to sample from posterior distribution and offers an insight into network's confidence.

After this retraining is finished, we perform inference by passing test samples a number of times from our network. Since the parameters of the last fully-connected layers of the network are sampled from a probability distribution, each

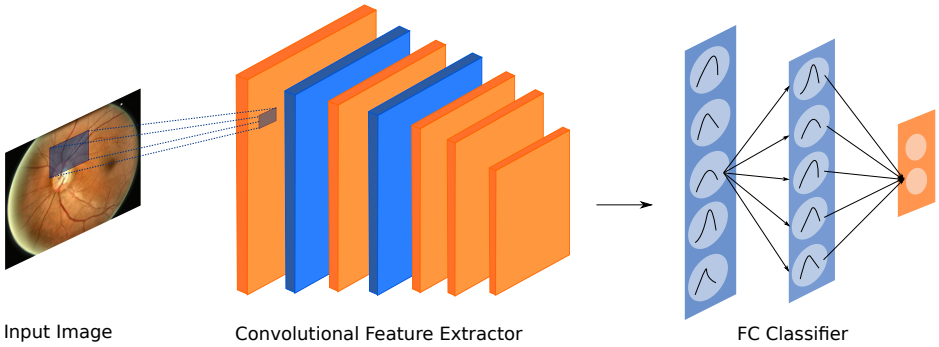


FIGURE 1: Proposed Hybrid Model. Convolutional Layers are trained separately using point estimates. The parameters of the convolutional layers are then frozen and Bayesian classifier is trained.

Algorithm 1 Uncertainty Estimation

Inputs $modelOutput$: Array containing softmax probabilities of all images for all models
$allPredictions$: Array containing class predictions for all images and for all models
$allTargets$: Array containing actual targets for all images and for all models
$percentile$: A scalar parameter to ascertain uncertain images to ignore
$consensus$: A scalar parameter representing minimum number of confident models to reach certain prediction
Outputs $certainAccuracy$: Accuracy when model is certain
$uncertainImages$: A percentage of uncertain images filtered out

procedure ESTIMATEUNCERTAINTY
  for each model i in $allModels$ **do**
    for each image j in $allImages$ **do**
      $differences$ = differences of top two classes' probabilities in $modelOutput[i][j]$
    end for
  end for
  $threshold$ = calculate for each model by filtering $percentile$ number of images from $differences$ of each model and average them.
  for each image j in $allImages$ **do**
    Let $confPred = 0$, $uncertain = 0$, $confModels = 0$ be new variables
    for each model i in $allModels$ **do**
      if $differences[i][j] > threshold$ **then**
        if $allPredictions[i][j] == allTargets[i][j]$ **then**
          increment $confModels$
        end if
      end if
    end for
    if $confModels >= consensus$ **then**
      increment $confPred$
    else
      increment $uncertain$
    end if
  end for
  return $confPred/(len(allImages) - uncertain)$, $uncertain/len(allImages)$
end procedure

pass of the same test sample gives a different prediction. These output predictions are used to draw a posterior distribution and estimate network's uncertainty. Complete algorithms used for this task is given in Algorithm 1.

For uncertainty analysis in Bayesian and hybrid architectures during inference, the algorithm works by sampling 10 classifier models from Bayesian weights distribution for every test sample and taking their output predictions. This way, instead of a single prediction, we get a set of predictions representing a probability distribution on network's output. This set of predictions are normalised in $[0-1]$ range using min-max normalisation for direct comparison. Predictions for top two classes are taken and difference in their values is recorded. After having the normalised differences, we build a distribution of all these differences and use a percentile value (40% in this case) to automatically select a threshold for the measure of uncertainty. The percentile value of 40% is determined heuristically. This parameter can be considered as a knob to control how confident predictions are desired in any given application area. In circumstances where no prediction is deemed better than a wrong prediction (medical diagnosis, for example), this value can be raised to ensure that only the most confident predictions are given by the network. For other, relatively less critical, scenarios this knob can be adjusted accordingly. The underlying assumption for our uncertainty estimation is that if the output for two classes is fairly distinctive then the difference in top two classes should be greater than the threshold and the model is regarded as certain about prediction otherwise it is considered uncertain. If a test sample is regarded as certain by more than half models (represented by consensus parameter), using simple majority voting, then it is output as a fairly certain prediction.

A. TIME AND SPACE COMPLEXITY ANALYSIS

The proposed hybrid model uses fewer parameters than its Bayesian counterpart as is evident from Table 1. The table shows the number of trainable parameters in each method and training time per epoch for some of the datasets. The hybrid model does not incur any additional cost for combining the benefits of both deterministic and Bayesian methods.

The time complexity of the Algorithm 1 is $O(2M \times I)$, where M represents number of Models sampled and I denotes the number of test samples. Also, the algorithm computes in constant space since, regardless of number of total models and test samples, only one model and one test sample are loaded at any given time.

TABLE 1: Time and space requirement of deterministic, Bayesian and hybrid models for some datasets

Dataset	Network Parameters (Millions)			Execution Time per epoch (s)		
	Deterministic	Bayesian [18]	Hybrid [Ours]	Deterministic	Bayesian [18]	Hybrid [Ours]
MNIST	2.457	4.914	2.459	15	70	27
CIFAR-10	5.851	11.703	9.528	25	129	49
ISIC-Subset	58.294	116.587	112.840	338	832	602
ORIGA	58.29	116.579	112.831	5	16	6
Electric Devices	0.655	3.277	0.577	2	16	3
Mallat	3.801	33.423	3.486	2	10	3
Thorax-1	2.726	24.589	2.569	2	10	5

III. EXPERIMENTATION

We used 13 datasets of disparate modalities and from diverse areas of application to ascertain the viability of our proposed hybrid CNN architecture. A brief description of all the datasets used and overall experimental setup is given below.

A. DATASETS

Table 2 gives an overview of all the datasets used in this work. We picked standard benchmark image datasets, as well as challenging fine-grained medical image classification datasets and many time-series datasets so that the validity of our approach on a broad range of datasets may be extensively investigated.

TABLE 2: Distribution of datasets used to evaluate proposed architecture

Datasets	Modality	No. of Classes	No. of Samples		
			Train	Test	Total
Image Datasets					
MNIST	Grey Images	10	70k	10k	80k
CIFAR-10	Color Images	10	50k	10k	60k
Medical Image Datasets					
ORIGA	Color Retinal Fundus Images	2	520	130	650
ISIC-Subset	Color Clinical Skin Images	3	5201	600	5801
Time Series Datasets					
Fish	Image-derived data	7	175	175	350
ShapesAll	Image-derived data	60	600	600	1200
Plane	Sensor data	7	105	105	210
TwoPattern	Simulation data	4	1000	4000	5000
ECG5000	ECG data	5	500	4500	5000
MedicalImages	Image-derived data	10	381	760	1141
ElectricalDevices	Device data	7	8926	7711	16637
Mallat	Simulation data	8	55	2345	2400
ECG Thorax1	ECG data	42	1800	1965	3765

1) Image Datasets

We used two of the most common benchmark datasets i.e. MNIST [19] and CIFAR-10 [40] and two publicly available medical image datasets i.e. ORIGA [41] and a subset of ISIC Archive to evaluate the performance of our proposed approach. For MNIST and CIFAR-10, standard pre-defined train and test splits are used. ORIGA dataset provides clinical ground truth to benchmark segmentation of optic disc and classification of healthy and glaucomatous images. Since this dataset is very small and no predefined train and test splits are given, we used 5-fold Cross Validation (CV) for this dataset such that in each iteration of CV there are 130 images in validation fold and 520 images in training fold. The second dataset of medical images was taken from ISIC Archive 2018 version. It consists of around 24000 clinical and dermoscopic images of skin lesions categorised into 7 classes. Some of the classes in this dataset have as fewer as 122 images per class, therefore, we took a subset of the whole data with three largest classes namely Benign Keratosis-like Lesions (BKL), Melanoma (MEL), and melanocytic Nevi (NV) and randomly divided them into training and test sets.

2) Timeseries Datasets

We selected 9 datasets from UCR archive [42]. The time-series datasets were generated based on different sources including device usage, sensors data, ECG, motion sensor, or simulation etc. Each time-series contains different number of classes; and the number of observations also vary in each dataset. All datasets are already divided into train and test sets by the publisher.

B. PREPROCESSING

To preprocess benchmark image datasets (MNIST and CIFAR-10), we used random crop and normalisation by mean subtraction. On medical image datasets (ORIGA and ISIC Subset), histogram equalisation is applied to enhance contrast and normalize brightness. We also made use of different data augmentation techniques like rotations, flipping, and random crops to increase the dataset size. Note that in addition to preprocessed images, original images are also kept in the dataset. Data augmentation was done keeping in mind the class ratio, such that the minor class can have more augmentations and more copies generated. Time-series datasets are used without any preprocessing.

C. EXPERIMENTAL SETUP AND HYPERPARAMETER SELECTION

All of our image datasets were trained and compared with similar experimental setup. We used a 5-layer convolutional block as baseline CNN, however, our experiments with varying depths and breadths of CNN shows that the approach is fairly scalable to more advance CNN architectures. We trained this CNN using Maximum Likelihood Estimation (MLE) for 60 epochs with a learning rate of 0.001, weight decay of 5×10^{-4}, and batch size of 32. For probabilistic models, we used the same setup as described above but instead of using point estimates we trained convolutional and fully connected layers with distribution-based weights using Bayes by Backprop for 60 epochs. In our proposed hybrid approach, we employed a fully-connected classifier with frozen convolutional feature extractor, pre-trained using MLE, and fine-tuned it using Bayesian estimation for 60 epochs with similar parameters. Two hyperparameters used in Algorithm 1, i.e. $percentile$ and $consensus$ can be selected as per use case requirements. In critical application areas, for example medical image diagnosis or stock market prediction, where there is little room for incorrect classification, higher values of these parameters can be selected to ensure only the most certain predictions are given by the network. In other applications, a relaxed criterion for uncertainty estimation might be acceptable. In our experiments, we used $percentile = 40\%$ and $consensus$ of more than half models (i.e. 6 models). These values were selected empirically and they worked well in all 13 datasets of different kind. It should be emphasised here that, for a given dataset, we used the same underlying architecture (number, width, and depth of convolutional layers and size of dense layers) in all three training paradigms, i.e. fully deterministic, fully Bayesian and Hybrid, to ensure fair comparison among three approaches.

For time-series modality, we used CNN with two convolutional layers, each followed by a max pooling layer for deterministic model analysis. On top of that, two fully connected layers were added as classifier. For probabilistic and hybrid approach, we used the same setting as explained before.

IV. RESULTS AND ANALYSIS

Table 3 summarises classification accuracies obtained by traditional fully deterministic CNN, Bayesian CNN [18] and our proposed hybrid approach. The table shows that the proposed hybrid approach outperforms not only purely Bayesian CNNs but also their deterministic counterparts in 9 out of 13 datasets while giving comparable results on rest of them. Even when the hybrid approach lagged other methods in classification accuracies, the difference was very small and came at no additional cost in terms of time or number of parameters as shown in Table 1. The results in *Bayesian Accuracy* field in Table 3 are generated by our own experiments using the implementation of Shridhar et al. [18] for Bayesian CNNs.

TABLE 3: Comparison of deterministic, Bayesian, and proposed hybrid models on different datasets without using uncertainty estimation

Datasets	Deterministic Accuracy (%)	Bayesian [18] Accuracy (%)	Hybrid [Proposed] Accuracy (%)
Benchmark Datasets			
MNIST	99.0	99.01	**99.3**
CIFAR-10	88	72.0	**88.7**
Medical Image Datasets			
ORIGA	76	74.4	**80.3**
ISIC-Subset	74	65.5	**75.7**
Time Series Datasets			
Fish	**85.1**	80.7	84.7
ShapesAll	67.0	70.9	**72.3**
Plane	**97.0**	96.7	95.1
TwoPattern	89.0	81.0	**89.4**
ECG5000	92.0	**93.2**	91.9
MedicalImages	**69.0**	62.4	64.7
ElectricalDevices	55.0	54.0	**56.6**
Mallat	88.0	82.5	**89.3**
ECG Thorax1	90.0	89.1	**91.3**

Fig. 2 shows output probabilities of deterministic, Bayesian and hybrid models for various correctly classified and misclassified images from CIFAR-10 and ORIGA. It can be observed in Fig. 2 that when hybrid model was unable to make a correct prediction (subfigures (b), (d), (e), and (h)), it associated relatively smaller probability scores with its misclassification than its competing models who also misclassified but did so with overconfidence. Additionally, in cases where both deterministic and Bayesian models failed to correctly classify an image and hybrid network succeeded (subfigures (c), (f), and (g)), it predicted very cautiously with reasonable probability scores. The probability scores of hybrid model were at par with other two methods for relatively easy examples as shown in subfigure (a).

A. UNCERTAINTY ESTIMATION

Since deterministic model does not have intrinsic ability to estimate uncertainty (although some works like [24], [43] have used deterministic models and applied some post-processing to get confidence estimates), in this section we focus on Bayesian and Hybrid models only and compare their performance. Since the classifier part of both Bayesian and Hybrid methods are trained using Bayesian estimates, both networks provide posterior distribution which is used to estimate uncertainty using Algorithm-I. Table 4 compares the accuracies of both training methods before and after using Algorithm 1. In this table, *Overall Accuracy* refers the accuracy of the model before applying Algorithm 1, whereas *Certain Accuracy* refers to the accuracy on the predictions for which the network was certain according to Algorithm 1. When the algorithm is not sure about the prediction it tags the test sample as uncertain. We can observe that accuracies for both fully Bayesian and hybrid approaches improved after uncertainty estimation algorithm was applied.

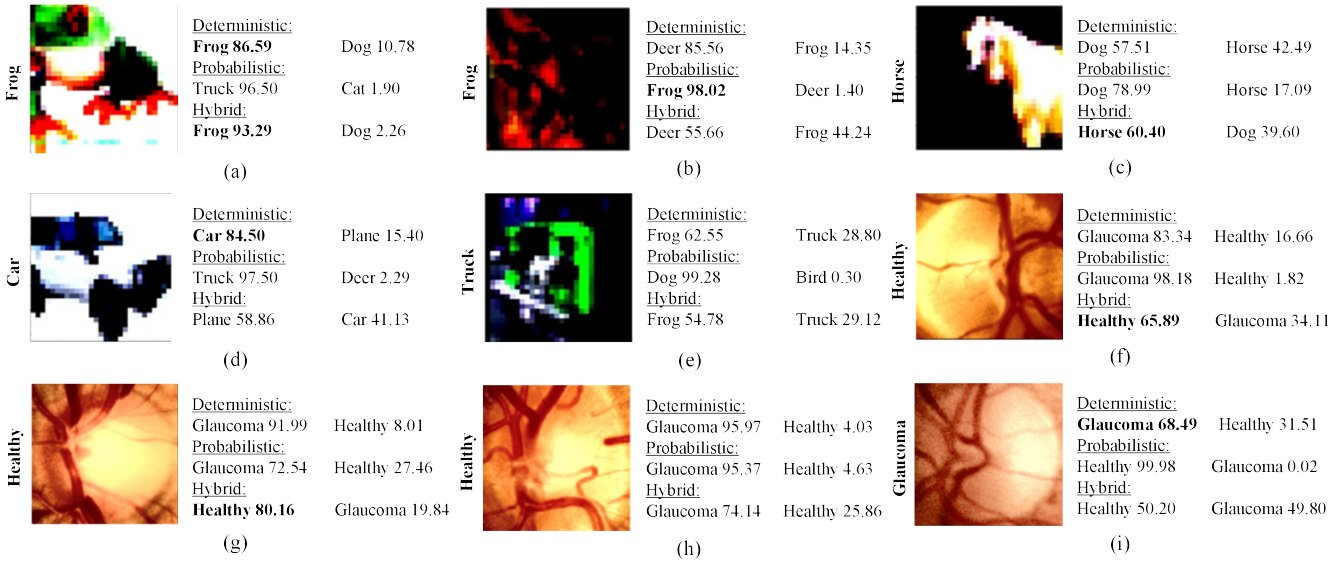


FIGURE 2: An analysis of confidence comparison for all three approaches on various samples of CIFAR10 and ORIGA datasets. The actual class is mentioned on left side of each image in bold vertical text.

TABLE 4: Comparison of Bayesian and proposed hybrid models on different datasets with uncertainty estimation

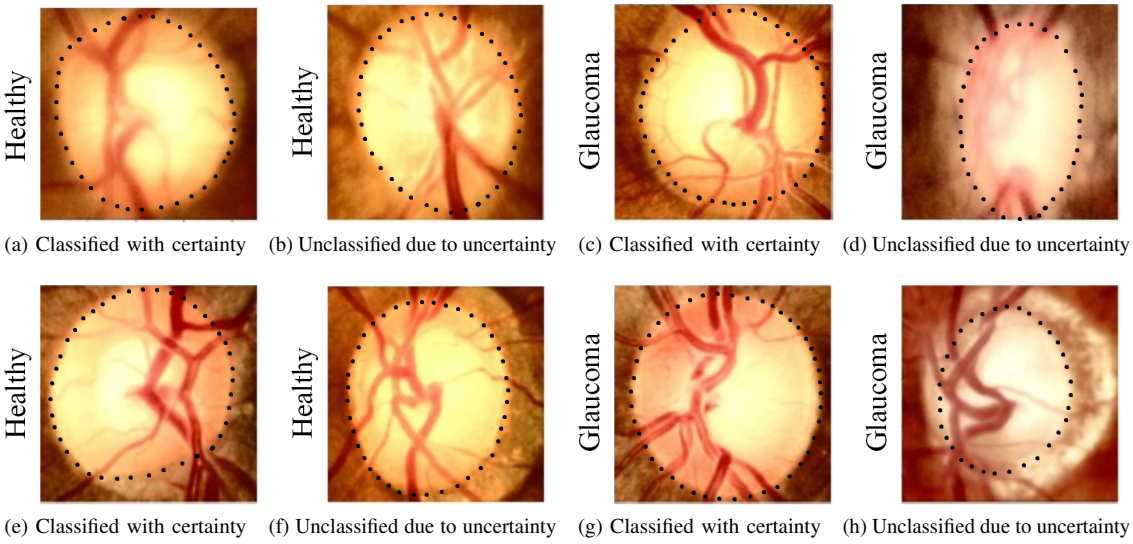
Datasets	Bayesian Model [18]			Hybrid Model		
	Overall Accuracy (%)	Certain Accuracy (%)	Uncertain Samples (%)	Overall Accuracy (%)	Certain Accuracy (%)	Uncertain Samples (%)
Image Datasets						
MNIST	99.01	99.17	20.5	**99.26**	**99.28**	9.6
CIFAR-10	65.41	72	66.9	**88.70**	**91.11**	46.2
Medical Image Datasets						
ORIGA	74.42	77.10	35.65	**80.31**	**77.21**	38.7
ISIC-Subset	58.15	65.48	34.3	**75.67**	**81.5**	53.8
Time Series Datasets						
Fish	80.7	92.4	9.1	**84.7**	**100.0**	6.8
ShapesAll	70.9	71.8	1.0	**72.3**	**72.9**	1.3
Plane	**96.7**	**98.9**	0.95	95.1	97.1	0.0
TwoPattern	81.0	84.4	25.0	**89.4**	**91.3**	24.9
ECG5000	**93.2**	93.8	36.2	91.9	**93.9**	36.8
MedicalImages	62.4	62.9	0.13	**64.7**	**66.5**	0.13
ElectricalDevices	54.0	55.8	14.6	**56.6**	**57.9**	14.8
Mallat	82.5	84.2	35.6	**89.3**	**92.1**	37.7
ECG Thorax1	89.1	90.9	14.9	**91.3**	**91.6**	14.8

The accuracy of our hybrid approach is higher than fully Bayesian model especially when it was fairly certain about the predictions. Fig. 3 shows some examples of images that were considered certain or uncertain by both Bayesian model (top row) and hybrid model (bottom row). It is very interesting to observe that the algorithm enabled both models to confidently categorised those images that had clearly defined optic disc border (black dotted elliptical boundary drawn on images to highlight disc boundary). In both training approaches the images where the boundary of the disc was dwindled, for examples because of papilledema (Fig. 3d and Fig. 3h) or optic atrophy (Fig. 3b and Fig. 3f), were filtered out and the model did not predict on these images because of high uncertainty.

Fig. 4 depicts the trade-off between number of uncertain samples and classification accuracy for both Bayesian and Hybrid models. We can see from this figure that the accuracy of the networks increases with the increase in percentage of uncertain samples. It can be argued from these curves that since, *difficult* samples have been passed over by the classifier and prediction is given for *easy* samples only, that is why we see a positive trend in accuracy with growing number of uncertain samples. However, in many crucial application areas, it is better to abstain from giving any half-cooked prediction than making a potentially costly mistake. In medical image analysis, for instance, such non-compulsive classifiers can reduce the workload of human experts by screening relatively easy disease patterns and allowing the physicians to focus their time and energy only on the most challenging of the cases.

V. CONCLUSION

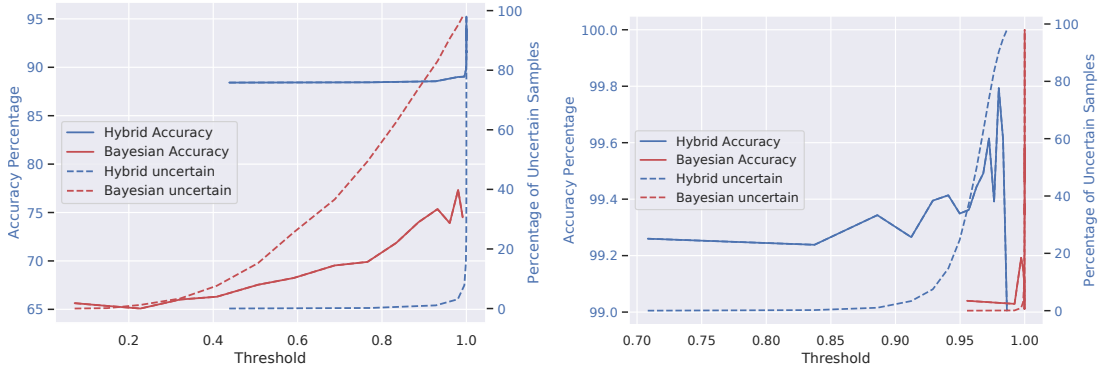
Practical applications of deep learning based classification models require high accuracy, better generalisation, computational efficiency and an estimate of uncertainty in model's predictions. All these characteristics are not readily available with either traditional deterministic CNNs or their Bayesian counterparts. Deterministic models, though provide better accuracies, do not facilitate uncertainty estimation on their own. Bayesian method, on the other hand, allows explication of posterior distribution but have significantly larger number of parameters that require more memory and time for tuning. Therefore, in this work we conceptualised and implemented a hybrid CNN capable of combining some of the merits of deterministic and Bayesian methods in terms of classification accuracy. The proposed method in validated on 13 different datasets and it shows promising results. We experimented with different architectures with varying number of convolutional and dense layers, and the hybrid training approach consistently performed better than its deterministic and Bayesian counterparts. We anticipate that this work might serves as a proof-of-concept that such hybrid CNN training is worth exploring since it works noticeably better than its pure deterministic and



(a) Classified with certainty (b) Unclassified due to uncertainty (c) Classified with certainty (d) Unclassified due to uncertainty

(e) Classified with certainty (f) Unclassified due to uncertainty (g) Classified with certainty (h) Unclassified due to uncertainty

FIGURE 3: Comparison of output probabilities for Bayesian and Hybrid training approaches on ORIGA dataset



(a) CIFAR-10 (b) MNIST

FIGURE 4: Trade-off between number of uncertain samples and the accuracy on remaining predictions. The threshold on x-axis is calculated using *percentile* parameter as shown in Algorithm 1.

probabilistic versions while at the same time facilitating estimation of network's certainty for every prediction. A thorough architecture search and hyper-parameter tuning might be required to increase baseline accuracies for each dataset. However, our experimentation with various data modalities and application areas has shown great promise to prompt further comprehensive investigation into this training paradigm. Our next logical step in this research would be to incorporate this hybrid approach with dataset specific architectures obtained through, for instance, NAS-Net [3] and ENAS [44] algorithms.

REFERENCES

[1] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in Advances in neural information processing systems, 2017, pp. 3856–3866.
[2] Y. Huang, Y. Cheng, D. Chen, H. Lee, J. Ngiam, Q. V. Le, and Z. Chen, "Gpipe: Efficient training of giant neural networks using pipeline parallelism," arXiv preprint arXiv:1811.06965, 2018.
[3] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 8697–8710.
[4] H. Haenssle, C. Fink, R. Schneiderbauer, F. Toberer, T. Buhl, A. Blum, A. Kalloo, A. B. H. Hassen, L. Thomas, A. Enk et al., "Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists," Annals of Oncology, vol. 29, no. 8, pp. 1836–1842, 2018.
[5] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," Nature, vol. 542, no. 7639, p. 115, 2017.
[6] H. Pratt, F. Coenen, D. M. Broadbent, S. P. Harding, and Y. Zheng, "Convolutional neural networks for diabetic retinopathy," Procedia Computer Science, vol. 90, pp. 200–205, 2016.
[7] X. Pei, "Emphysema classification using convolutional neural networks," in International Conference on Intelligent Robotics and Applications. Springer, 2015, pp. 455–461.
[8] M. N. Bajwa, M. I. Malik, S. A. Siddiqui, A. Dengel, F. Shafait, W. Neumeier, and S. Ahmed, "Two-stage framework for optic disc localization and glaucoma classification in retinal fundus images using deep learning," BMC medical informatics and decision making, vol. 19, no. 1, p. 136, 2019.
[9] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," nature, vol. 521, no. 7553, p. 436, 2015.
[10] Y. Zheng, Q. Liu, E. Chen, Y. Ge, and J. L. Zhao, "Time series classification using multi-channels deep convolutional neural networks," in International Conference on Web-Age Information Management. Springer, 2014, pp. 298–310.
[11] M. Munir, S. A. Siddiqui, A. Dengel, and S. Ahmed, "Deepant: A deep learning approach for unsupervised anomaly detection in time series," IEEE Access, vol. 7, pp. 1991–2005, 2019.
[12] D. Ho, E. Liang, I. Stoica, P. Abbeel, and X. Chen, "Population based augmentation: Efficient learning of augmentation policy schedules," arXiv preprint arXiv:1905.05393, 2019.
[13] K. Shridhar, A comprehensive guide to Bayesian CNN with variational inference : with implementation in PyTorch. LAP Lambert Academic Publishing, 2019.
[14] J. Snoek, O. Rippel, K. Swersky, R. Kiros, N. Satish, N. Sundaram, M. Patwary, M. Prabhat, and R. Adams, "Scalable bayesian optimization using deep neural networks," in International conference on machine learning, 2015, pp. 2171–2180.
[15] R. M. Neal, "Probabilistic inference using markov chain monte carlo methods," 1993.
[16] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An introduction to variational methods for graphical models," Machine learning, vol. 37, no. 2, pp. 183–233, 1999.
[17] T. Salimans, D. P. Kingma, and M. Welling, "Markov chain monte carlo and variational inference: Bridging the gap," arXiv preprint arXiv:1410.6460, 2014.
[18] K. Shridhar, F. Laumann, A. Llopart Maurin, and M. Liwicki, "Bayesian convolutional neural networks," arXiv preprint arXiv:1806.05978, 2018.
[19] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner et al., "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.
[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, 2012, pp. 1097–1105.
[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," CoRR, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385
[22] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, "Variational inference: A review for statisticians," Journal of the American Statistical Association, vol. 112, no. 518, pp. 859–877, 2017.
[23] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural networks," arXiv preprint arXiv:1505.05424, 2015.
[24] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in international conference on machine learning, 2016, pp. 1050–1059.
[25] Y. Kwon, J.-H. Won, B. J. Kim, and M. C. Paik, "Uncertainty quantification using bayesian neural networks in classification: Application to ischemic stroke lesion segmentation," 2018.
[26] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" in Advances in neural information processing systems, 2017, pp. 5574–5584.
[27] Y. Tang and R. R. Salakhutdinov, "Learning stochastic feedforward neural networks," in Advances in Neural Information Processing Systems, 2013, pp. 530–538.

[28] M. Lázaro-Gredilla and A. R. Figueiras-Vidal, "Marginalized neural network mixtures for large-scale regression," IEEE transactions on neural networks, vol. 21, no. 8, pp. 1345–1351, 2010.
[29] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," arXiv preprint arXiv:1610.02136, 2016.
[30] K. Lee, H. Lee, K. Lee, and J. Shin, "Training confidence-calibrated classifiers for detecting out-of-distribution samples," arXiv preprint arXiv:1711.09325, 2017.
[31] G. Papamakarios, T. Pavlakou, and I. Murray, "Masked autoregressive flow for density estimation," in Advances in Neural Information Processing Systems, 2017, pp. 2338–2347.
[32] A. v. d. Oord, Y. Li, I. Babuschkin, K. Simonyan, O. Vinyals, K. Kavukcuoglu, G. v. d. Driessche, E. Lockhart, L. C. Cobo, F. Stimberg et al., "Parallel wavenet: Fast high-fidelity speech synthesis," arXiv preprint arXiv:1711.10433, 2017.
[33] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," in Advances in neural information processing systems, 2016, pp. 4743–4751.
[34] C.-W. Huang, D. Krueger, A. Lacoste, and A. Courville, "Neural autoregressive flows," arXiv preprint arXiv:1804.00779, 2018.
[35] N. De Cao, I. Titov, and W. Aziz, "Block neural autoregressive flow," arXiv preprint arXiv:1904.04676, 2019.
[36] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," arXiv preprint arXiv:1605.08803, 2016.
[37] L. Ardizzone, J. Kruse, S. Wirkert, D. Rahner, E. W. Pellegrini, R. S. Klessen, L. Maier-Hein, C. Rother, and U. Köthe, "Analyzing inverse problems with invertible neural networks," arXiv preprint arXiv:1808.04730, 2018.
[38] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," in Advances in Neural Information Processing Systems, 2018, pp. 10 215–10 224.
[39] L. Dinh, D. Krueger, and Y. Bengio, "Nice: Non-linear independent components estimation," arXiv preprint arXiv:1410.8516, 2014.
[40] A. Torralba, R. Fergus, and W. T. Freeman, "80 million tiny images: A large data set for nonparametric object and scene recognition," IEEE transactions on pattern analysis and machine intelligence, vol. 30, no. 11, pp. 1958–1970, 2008.
[41] Z. Zhang, F. S. Yin, J. Liu, W. K. Wong, N. M. Tan, B. H. Lee, J. Cheng, and T. Y. Wong, "Origa-light: An online retinal fundus image database for glaucoma analysis and research," in 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology. IEEE, 2010, pp. 3065–3068.
[42] H. A. Dau, E. Keogh, K. Kamgar, C.-C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, Yanping, B. Hu, N. Begum, A. Bagnall, A. Mueen, and G. Batista, "The ucr time series classification archive," October 2018, https://www.cs.ucr.edu/~eamonn/time_series_data_2018/.
[43] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017, pp. 1321–1330.
[44] H. Pham, M. Y. Guan, B. Zoph, Q. V. Le, and J. Dean, "Efficient neural architecture search via parameter sharing," arXiv preprint arXiv:1802.03268, 2018.

MUHAMMAD NASEER BAJWA completed his BS in Computer Engineering from COMSATS Institute of Information Technology (CIIT), Pakistan and MS in Computer Engineering from King Fahd University of Petroleum and Minerals (KFUPM), Saudi Arabia. He is presently pursuing PhD from Technische Universität Kaiserslautern, Germany and is also working as Research Assistant at German Research Center for Artificial Intelligence GmbH (DFKI). His main area of research is towards realising a practically usable, confident and interpretable Computer-Aided Diagnosis (CAD) system. He has published his works on detection of ocular disorders like glaucoma and diabetic retinopathy using retinal fundus images, automated diagnosis of cutaneous diseases using dermoscopic images, curation of retinal fundus images dataset for glaucoma detection (G1020), and interpretability of CAD for skin lesions.

SULEMAN KHURRAM received his B.S. Computer Science from Arid Agriculture University, Pakistan in 2014 and is currently doing M.Sc. Computer Science with specialisation in Intelligent Systems from Technische Universität Kaiserslautern, Germany. He had worked as Software Developer Jin Technologies, Pakistan and in 2015 he joined Next Controls, United Kingdom as Java Consultant. He also worked as Research Assistant in Research Assistant at German Research Center for Artificial Intelligence GmbH (DFKI), Germany. Currently he is actively managing and handling two products of Checkit United Kingdom as Consultant, where he deals with development, management and support for energy services backend.

MOHSIN MUNIR received his master's degree in Computer Science from Technical University of Kaiserslautern, Germany. He did internships at RICOH (Japan) and BOSCH (Germany) during his master's degree. The topic of his master's thesis was 'Connected Heating System's Fault Detection using Data Anomalies and Trends'. Currently, he is pursuing his Ph.D. in Computer Science at German Research Center for Artificial Intelligence GmbH (DFKI) under the supervision of Prof. Dr. Prof. h.c. Andreas Dengel. His research topic is 'Time Series Forecasting and Anomaly Detection'. His research interests are time series analysis, deep neural networks, forecasting, predictive analytics, and anomaly detection. During his Ph.D., he did a research internship at Kyushu University (Japan) under the supervision of Prof. Seiichi Uchida.

SHOAIB AHMED SIDDIQUI is currently pursuing MS leading to Ph.D. program at German Research Center for Artificial Intelligence GmbH (DFKI) and Technische Universität Kaiserslautern under supervision of Prof. Dr. Prof. h.c. Andreas Dengel. He received his Bachelor's degree in Computer Science from National University of Sciences and Technology (NUST), Pakistan in 2016. His areas of interest include interpretability and robustness of deep learning models (including adversarial examples and defenses), document understanding, time series analysis, extreme classification and reinforcement learning. He is also a reviewer for ICES Journal of Marine Science and IEEE Access.

MUHAMMAD IMRAN MALIK received his master's and PhD degrees in Artificial Intelligence, in 2011 and 2015 respectively, from the University of Kaiserslautern. He also worked in the German Research Center for Artificial Intelligence GmbH (DFKI), Kaiserslautern, Germany. His Ph.D. topic was automated forensic handwriting analysis on which he focused on both the perspectives of forensic handwriting examiners and pattern recognition researchers. He is currently an Assistant Professor with the School of Electrical Engineering and Computer Science (SEECS) at the National University of Sciences and Technology (NUST), Islamabad, Pakistan. He has authored more than 40 publications including several journal and high ranked conference papers.

ANDREAS DENGEL is Scientific Director at German Research Center for Artificial Intelligence GmbH (DFKI) in Kaiserslautern. In 1993, he became a Professor at Computer Science Department of the University of Kaiserslautern where he holds the chair Knowledge-Based Systems. Since 2009 he is appointed Professor (Kyakuin) at the Department of Computer Science and Information Systems at the Osaka Prefecture University. He received his Diploma in CS from the University of Kaiserslautern and his PhD from the University of Stuttgart. He also worked at IBM, Siemens, and Xerox Parc. Andreas is member of several international advisory boards, chaired major international conferences, and founded several successful start-up companies. Moreover, he is co-editor of international computer science journals and has written or edited 12 books. He is author of more than 300 peer-reviewed scientific publications and supervised more than 170 PhD and master theses. Andreas is a IAPR Fellow and received many prominent international awards. His main scientific emphasis is in the areas of Pattern Recognition, Document Understanding, Information Retrieval, Multimedia Mining, Semantic Technologies, and Social Media.

SHERAZ AHMED is Senior Researcher at German Research Center for Artificial Intelligence GmbH (DFKI) in Kaiserslautern, where he is leading the area of Time Series Analysis. He received his master's degree in Computer Science from Technische Universität Kaiserslautern, Germany. He completed his PhD in the German Research Center for Artificial Intelligence, Germany, under the supervision of Prof. Dr. Prof. h.c. Andreas Dengel and Prof. Dr. habil. Marcus Liwicki. His PhD topic is Generic Methods for Information Segmentation in Document Images. Over the last few years, he has primarily worked on development of various systems for information segmentation in document images. His research interest includes document understanding, generic segmentation framework for documents, gesture recognition, pattern recognition, data mining, anomaly detection, and natural language processing. He has more than 30 publications on the said and related topics including three journal papers and two book chapters. He is a frequent reviewer of various journals and conferences including Patter Recognition Letters, Neural Computing and Applications, IJDAR, ICDAR, ICFHR, and DAS. From October 2012 to April 2013 he visited Osaka Prefecture University (Osaka, Japan) as a research fellow, supported by the Japanese Society for the Promotion of Science and from September 2014 to November 2014 he visited University of Western Australia (Perth, Australia) as a research fellow, supported by the DAAD, Germany and Go - 8, Australia.