Transfer Learning Enabled Convolutional Neural Networks for Cutting Tool Prognostics

Mohamed Marei – Year 3 PhD Student
Director of Studies: Weidong Li
Second Supervisor: Nazaraf Shah
Third Supervisor: Kuo-Ming Chao

Abstract

Cutting tool Prognostics and Health Management (PHM) is vital for manufacturing quality assurance, process waste reduction, and maximising machine availability. The challenge of reliably predicting tool health, and consequently, tool remaining useful life (RUL) is complicated due to the variation of cutting parameters used to machine a part or component. To address this limitation and, furthermore, maximize the use of limited examples of collected visual inspection data, we present a data-driven deep learning approach to transfer image feature recognition abilities from Convolutional Neural Network (CNN) models, initially trained for general-purpose image category classification. By enforcing a penalty on the transfer of task-irrelevant image features, our transfer learning approach learns to output tool health (as regression) through analysing imbalanced examples of healthy and worn tools, with an accuracy of up to 84%, and can utilize several popular pre-trained CNN architectures to learn features descriptive of the new task objective.

Motivation

Cutting tools in CNC machining undergo tool wear, which reduces their lifetime. Reducing the number of tools replaced can significantly reduce the cost per workpiece in terms of tool change time and tool costs. Accurate prognosis of the remaining useful life (RUL) of the tool can therefore reduce unnecessary replacements.

Problem

Accurate cutting tool prognosis is challenging due to variability in the machining parameters used to model the tool life.

Objectives

- To develop a parameter-free approach for predicting machine tool remaining useful life (RUL) from limited visual inspection images.
- To leverage the image analysis capabilities of Deep Learning models to predict tool wear from limited visual inspection images, with no knowledge of machining parameters.
- To explicitly reduce negative transfer, arising from re-using CNN model parameters which have been trained on massive general-purpose classification data (i.e. source task), in predicting the outputs for the target task.

Methodology

In this research, an approach to leverage pre-trained CNN models to predict cutting tool health has been applied. Firstly, the CNNs are modified with additional task-specific layers to learn new features distilled from previous network layers. The images of cutting tools are provided as input into the CNNs, which learn to directly output the tool flank wear width (in mm).

To reduce the impact of negative transfer, the learned feature representation of the source task and target task is minimized directly. This is achieved by adding an objective to the loss function of the CNN, where the source and target feature distributions means are contrasted. This quantity, termed the Maximum Mean Discrepancy (MMD), has been previously proposed to evaluate the similarity between two probability distributions, and has been found to considerably reduce the effects of negative transfer. In this research the MMD distance is incorporated to the model's objective risk function, which becomes a weighted summation of the root mean square error (RMSE) and the MMD. The six pre-trained architectures and their classification performance is summarized in table 1.

Results and Analysis

- Source fine-tuning then incorporating the MMD loss (Method 1) improves prediction accuracy by ~5.34% (83.97%) from source fine-tuning alone (Method 2) (78.63%), yielding lower RMSE for the best performing method (0.1565 mm vs 0.163). Both methods outperformed full re-training (Method 3) in terms of accuracy (77.2%) and RMSE (0.183).
- Source fine-tuning method 2 took less time than methods 1 and 3 (~7550 s), with method 1 being the slowest overall to reach a RMSE of 0.2. This is due to the computational expense of the MMD calculation. However, the improvements are considerable in terms of prediction accuracy.

Conclusions and Future Works

The approach discussed in this work can improve the predictive power of CNNs, making them more capable at learning different visual concepts from those which they have been designed for. This demonstrates the flexibility of CNNs in learning both generic and domain-specific features, subject to a principled application of the MMD as a loss constraint. Despite the temporal dependencies present in the data, the approach presented in this work can considerably reduce overfitting of CNN on new data. We plan to test our approach in conjunction with recurrent neural networks (RNN) to learn time dependencies of the observations and targets. We also plan to evaluate prediction accuracy by incorporating the machining parameters into the inputs.

Dataset

The dataset comprises 553 images of cutting tools captured at given intervals of cutting length. Each image has been assigned a flank wear width value (in mm), along with the cutting distance (in metres).

- First a pre-processing operation is used to crop the images of size 1600x1200 pixels into 801x801 pixels, which are further reduced to the desired network input size.
- Due to the limited number of images, data augmentation is performed on the images at training time, where random translation, rotation and scaling operations are applied to the images.
- The data is divided into 80% (441 samples) for training and 20% (112) for validation.

Experiments

Each model is trained for 750 epochs, with a mini-batch size of 16 images. The learning rate for the first 2 methods is set to $4x10^{-5}$, and $4x10^{-4}$ for the third method. The Three experiments have been carried out with each model:

- Source fine-tuning with MMD computation (Method 1): the network is first trained with the source domain dataset and task then the final layer removed and the model re-trained with the MMD layer (FT_MSE_MMD), i.e. our approach.
- Source fine-tuning with no MMD computation (Method 2): the network is first trained on the source domain dataset and task, then re-trained on the target domain data and task without MMD (FT_MSE_noMMD).
- Full re-training of network from scratch (Method 3) (No_FT_noMMD).

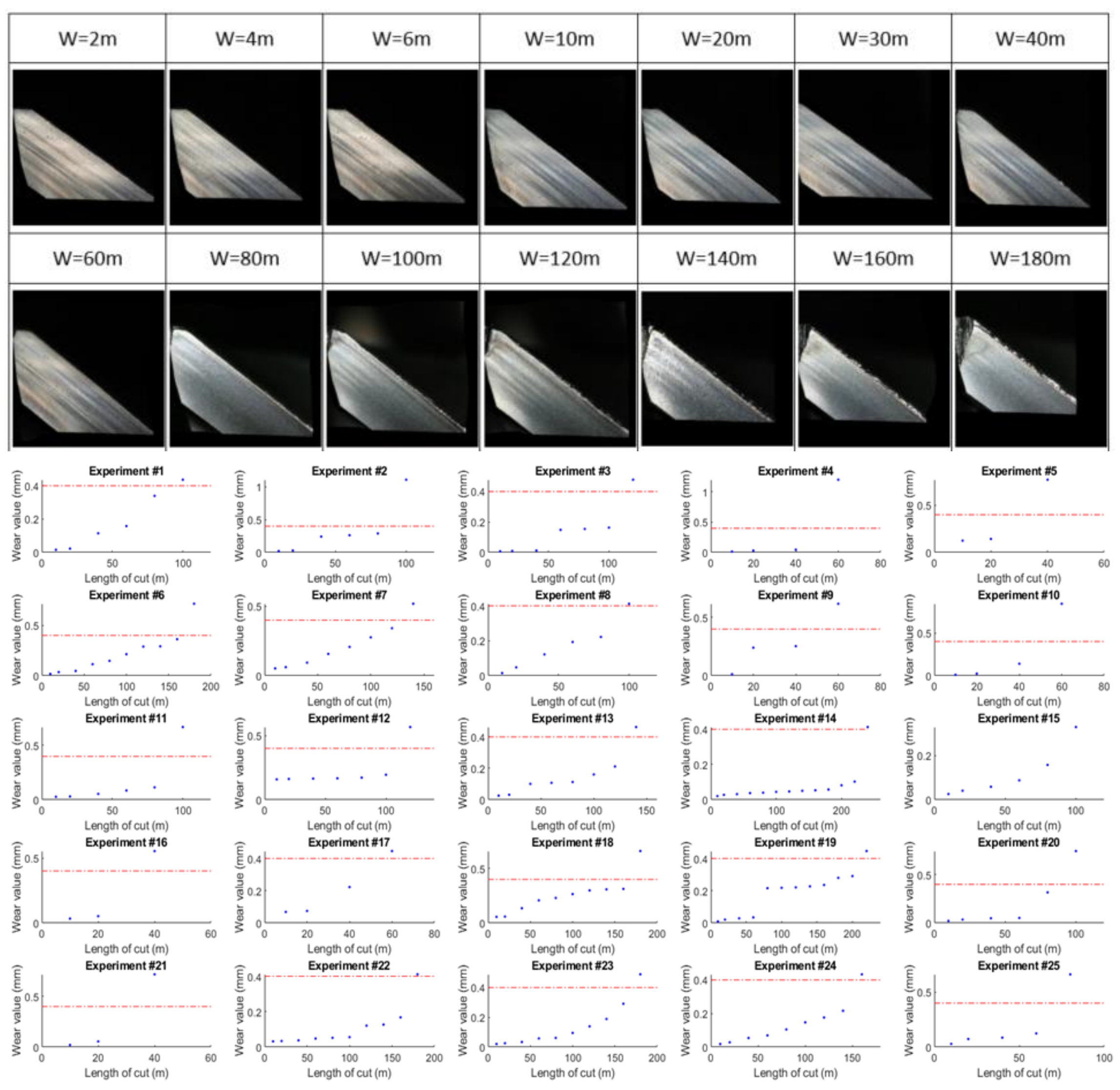

Figure 1. a) Experiment 1 samples, showing the pre-processed tool wear images at increasing cutting tool lengths (in m), b) cutting tool wear recorded vs length of cut for 25 experiments.

Table 1. Summary of CNN accuracy for general-purpose image classification

Network [Reference]	Top-1 Accuracy (%)	Top-5 Accuracy (%)	Parameters (Millions)	Image Input Size
AlexNet [2]	63.3	84.6	61.0	227×227
ResNet-18 [3]	71.78	90.58	11.7	224×224
ResNet-50 [3]	77.15	93.29	25.6	224×224
ResNet-101 [3]	78.25	93.95	44.6	224×224
SqueezeNet [4]	60.4	82.50	1.24	227×227
InceptionV3 [5]	78.95	94.49	23.9	299×299

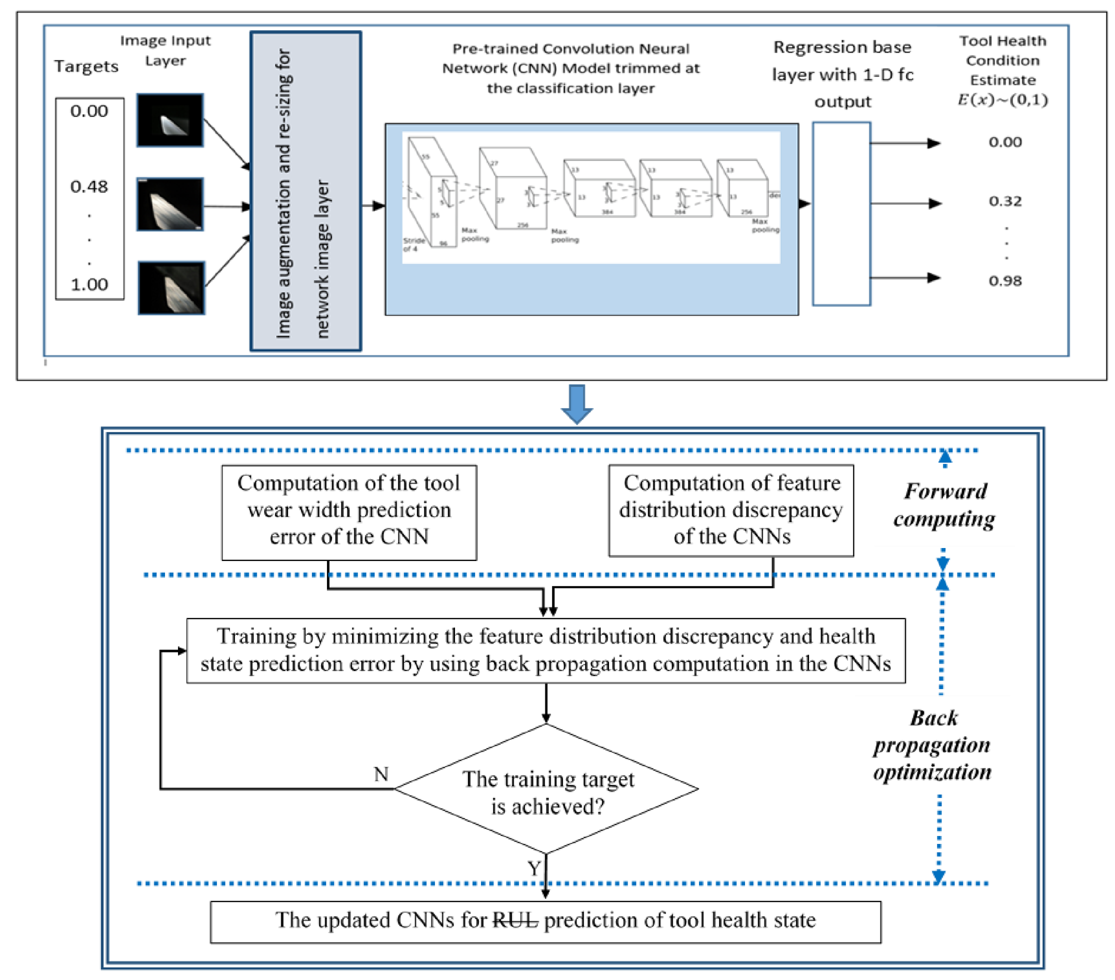

Figure 2. Method 1 (our approach) for transfer learning CNN tool wear prediction

Table 2. Summary of Validation results for CNN tool wear prediction

Pre-trained Model Name	Training Details	Model Performance (Validation Dataset)					
	Training Method	MAE	RMSE	acc_{10} (%)	acc_{20} (%)	acc_{30} (%)	Training Time (s)
AlexNet [2]	**Method1**	**0.0829**	**0.1684**	**81.68**	**90.84**	**91.60**	2125
	Method 2	0.0868	0.1723	79.39	90.08	**91.60**	**1940**
	Method 3	0.0903	0.1726	77.86	90.08	90.84	2028
ResNet-18 [3]	**Method 1**	**0.0773**	0.1654	**83.97**	**90.84**	**92.37**	3358
	Method 2	0.0820	**0.1591**	78.63	90.08	**92.37**	**2649**
	Method 3	0.0791	0.1594	80.15	90.08	**92.37**	2854
ResNet-50 [3]	Method 1	**0.0868**	**0.1764**	**80.92**	**86.26**	**90.84**	14790
	Method 2	0.1124	0.1967	74.05	84.73	90.08	**9184**
	Method 3	0.1050	0.1954	76.34	83.97	90.08	12689
ResNet-101 [3]	Method 1	**0.0833**	0.1657	74.81	**89.31**	92.37	17750
	Method 2	0.0992	**0.1565**	74.05	**89.31**	**93.89**	**15122**
	Method 3	0.0882	0.1740	**77.86**	86.26	90.84	16654
SqueezeNet [4]	Method 1	0.0891	**0.1732**	**79.39**	88.55	**91.60**	2152
	Method 2	**0.0868**	0.1784	**79.39**	**89.31**	**91.60**	**1764**
	Method 3	0.0882	0.1710	77.68	88.55	91.60	1878
InceptionV3 [5]	Method 1	**0.0886**	0.1784	**79.39**	85.50	**91.60**	20334
	Method 2	0.1040	0.1931	77.10	85.50	90.08	**14653**
	Method 3	0.0916	**0.1728**	**79.39**	**87.02**	**91.60**	18780
Average	**Method 1**	**0.0847**	**0.1713**	**80.03**	**88.55**	**91.73**	10085
	Method 2	0.0952	0.1760	77.10	88.17	91.60	**7552**
	Method 3	0.0904	0.1742	78.21	87.66	91.22	9147

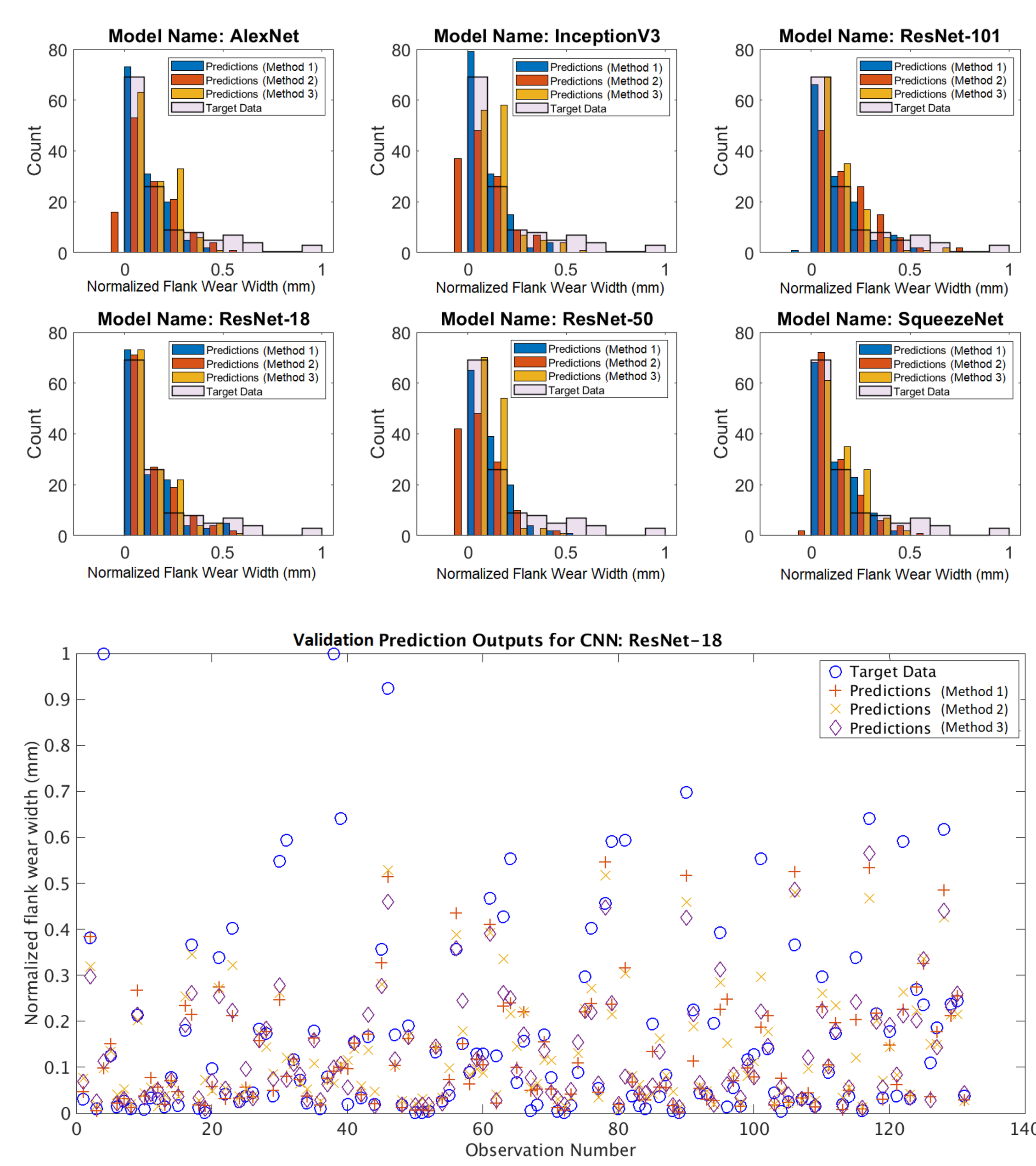

Figure 3a. Prediction histograms (top); 3b. sample outputs for ResNet-18 (bottom)

References

[1] Deng, J., et al. "Large scale visual recognition challenge 2012." *ILSVRC 2012 Workshop*. 2012.
[2] Krizhevsky et al. AlexNet. 2012.
[3]K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 770–778, Jun. 2015.
[4] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size," *Int. Conf. Comput. Vis. Pattern Recognit.*, Feb. 2016.
[5] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Sep. 2014.
[6] Yan, Hongliang, et al. "Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.