Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 00 (2021) 000–000

www.elsevier.com/locate/procedia

55th CIRP Conference on Manufacturing Systems

ConvLSTM deep learning signal prediction for forecasting bending moment for tool condition monitoring

Stephanie Hall[a,]*, Stephen T. Newman[a], Evripides Loukaides[a], Alborz Shokrani[a]

[a]*Department of Mechanical Engineering, University of Bath, Bath, BA2 7AY*

* Corresponding author. Tel.: +44-1225-386588. *E-mail address:* sjh224@bath.ac.uk

Abstract

This paper presents a new method using a Convolutional Long Short-Term Memory (ConvLSTM) network to predict the future sensor signals for cutting tool bending moment in milling. Tool bending moment is directly correlated to cutting forces and can be used for tool condition monitoring. Bending moment from a sensory tool holder was obtained and transformed into uniform sized time-frequency domain data to be used as frame inputs to the network. Future frames of sensor signals are predicted and validated by comparing the predicted frames with the ground truth. The investigations show the potential of the proposed method for predicting future signals which can be used for tool condition monitoring.

Keywords: Deep learning; End milling; machining; Sensors

1. Introduction

Real-time monitoring of the status of the cutting tool has great significance; cutting tool and workpiece wastage and machine downtime can be reduced, thus reducing economic losses [1]. Being able to closely monitor the status of the cutting tool and progression of the wear is extremely beneficial to machining operators and has gained significant attention in recent years.

The domain of predictive analytics involves the use of statistical techniques and other predictive algorithms to predict the future by examining past and present time series data. With the increased volume and quality of data, more complex and efficient models need to be developed. Traditionally, physical models of systems have been used to provide predictions on the remaining useful life of different systems. However, these are not feasible for complex systems [2].

For predictions/classification involving time series data, deep learning architectures are becoming increasingly popular and the performance is improving [3]. Recurrent Neural Networks (RNNs), specifically Long Short-Term Memory (LSTM), are mainly used for problems involving sequence prediction. Examples of sequence prediction problems are; language translation, predicting future frames of a video and weather forecasting [4].

Recent studies within the domain of predictive algorithms and machine learning have often used manual feature extraction and reduction for the inputs to these algorithms. Whilst effective, handcrafted approaches rely on expert knowledge on the signal source and correct filtering techniques for the signal type [5]. Use of these methods is labour intensive and time consuming. Manually selecting the wrong features can influence the effectiveness of the model since extracted features directly determine the performance of the classification or prediction models. Deep learning is capable of providing a new approach to operate directly on the raw data and automatically learn and extract features without needing specific domain knowledge [6].

One of the most prominent frameworks for deep learning, particularly for classification problems, is the Convolutional Neural Network (CNN), which has been used in cases for tool condition monitoring/remaining useful life prediction [7], [8]. Although, some information may be lost between time steps as sequential and temporal dependency are not considered as data

is not treated as time series but rather as static spatial data. To deal with time series data, RNNs, specifically LSTM networks, are popular as they can capture long-term dependencies and nonlinear dynamics of time series data [9].

Deep learning methods have been widely used for classification and identification problems within the domain of manufacturing. Petruschke et al. [10] compared the use of CNN and RNN to identify energy states for machine tools; achieving over 95% accuracy for each method classifying states into labels ‘standby’, ‘operational’, and ‘working. Hu et al. [11] used a combination CNN-LSTM model to predict energy consumption in additive manufacturing; using slices from the CAD models to generate images at each layer.

For quality assessment within resistance spot welding, Stavropoulos et al. [12] propose an online monitoring system using video data obtained from an infrared camera. They test a variety of machine learning classification models. Papacharalampopoulos et al. [13] integrate an LSTM model into their online monitoring system Digital Twin for laser powder bed fusion.

Cutting forces have been identified as the direct predictor of machining conditions including tool wear [14]. Being able to predict cutting forces in a time series scenario allows for monitoring cutting tool condition and the workpiece precision.

In this paper, ConvLSTM models are applied to time series data obtained from a sensory tool holder that records the tool bending moments in two dimensions during machining. The aim is to be able to predict the future signals from the sensors based on an input sequence.

The following section describes the deep learning model used to predict frames for the future bending moment signal. The methodology for data collection and analysis is described in section 3. In sections 4 and 5 the results are presented and discussed in detail followed by conclusions and future work in section 6.

2. Deep learning for series prediction

Recurrent neural networks are specialised networks used to process sequential or time series data of varying length, e.g. in the format of a video, audio, language, etc. [15]. The main feature of using these networks is that the structure can store and memorise historical information about the inputs. They essentially operate as a loop, where information is passed from each step to the next. However, RNNs do run into a problem with exploding or vanishing gradients, where if the gradient is too small, the weight parameter updates tend towards zero, and if it is too large, the model weights will become too large and eventually become ‘Not a Number (NaN)’ [3].

In order to address the problem of exploding or vanishing gradient in Recurrent Neural Networks (RNNs), the Long Short-Term Memory (LSTM) network was introduced [16].

In comparison to standard RNNs, LSTMs are capable of learning long-term dependencies within the data. The LSTM layer consists of multiple connected LSTM cell blocks. Compared to RNNs, LSTMs have a much more complex structure within the LSTM module, composed of various connections and gates that allow information to be regulated and flow through. The forget gate, input gate, and output gate can decide which information is forgotten, added, and passed along within the LSTM module, therefore ensuring relevant information moves along time steps to make predictions. The Convolutional LSTM (ConvLSTM) was proposed by Shi et al. [4] for the task of precipitation nowcasting. In this architecture, the matrix multiplication within the LSTM cell is replaced with a convolutional operation.

CNNs are best suited for working with image inputs. LSTMs are good at working with time series data as these networks can hold onto information from previous time steps and use this for decision making. However, they are not ideal for working with inputs that have a spatial structure. For sequential images that are spatiotemporal, such as frames in a video, a model that combines the two and can find both spatial and temporal correlations is ideal. As the multiplication within the LSTM cell is replaced with a convolutional operation in the ConvLSTM cell, the output keeps the same spatiotemporal dimension as the input, rather than outputting 1D feature vectors.

The input for a ConvLSTM has five dimensions; samples, time steps, channels, rows, and columns. The general structure of a ConvLSTM model includes combinations of ConvLSTM layers. As it has multiple stacked ConvLSTM layers, it has a strong representational power which makes it suitable for giving predictions in complex dynamical systems [4]. Typically, after a ConvLSTM layer, the data is fed through a batch normalisation layer. The output for this model is a combination of both an LSTM and a convolution layer output and has the same dimension as the input.

3. Methodology

3.1. Data collection

The dataset for this model was obtained from a set of machining experiments performed on a Bridgeport vertical machining centre (VMC 610XP). In total, six machining experiments were performed and cutting tool bending moment was collected using a SPIKE® wireless sensory tool holder at a frequency of 2.5 kHz. The experiments were end milling straight cuts along the length of a 50 × 50 × 150 mm block of Inconel 718 using climb milling strategy and flood cooling condition. Six uncoated cemented tungsten carbide end milling tools with 12 mm diameter end mills and varying cutting geometry were used. Two levels of helix angle (30° and 45°) and three levels of rake angles (4°, 8° and 12°) were used for a full factorial design of experiments. The machining parameters were 60 m/min cutting speed and 0.03 mm/tooth feed per tooth with 3 mm radial depth of cut and 1 mm axial depth of cut. Uncoated tools allow for accelerated tool wear which affects the tool bending moment and hence the cutting forces. The tool wear was monitored periodically, and the experiments were stopped once tool flank wear, V_B, reached 300 µm.

3.2. Data preprocessing

Previous work has shown that the tool bending moment is affected by the tool wear and can potentially be used for tool wear monitoring [17]. For the input data to the ConvLSTM

network, the bending moment in both the X and Y directions are combined into one signal by taking the absolute bending moment using the Pythagorean theorem. The resulting signals are then split into samples of 0.5 seconds which are then each normalised locally between 0 and 1. This will remove the impact of signal amplitude and the overall signal trend over the tool lifetime allowing for generalization of the method.

The main methods for signal processing and feature extraction are time domain, frequency domain, and time-frequency domain. Here time-frequency domain analysis is used in the form of Continuous Wavelet Transform (CWT). This enables transforming the signal sections into a matrix which can be represented as an image to be used as the input for the ConvLSTM network. This preprocessing step involves elements of feature selection prior to the analysis within the ConvLSTM network, where further feature engineering is performed on the generated images.

Fig. 1 illustrates the bending moment signal for three passes of the cutting tool. The corresponding continuous wavelet transforms for each signal is taken from a 0.5 second section in the middle of the pass which has been locally normalized. The signal from a new tool is shown in row (a) whilst the signal from a tool with ~300 µm wear is provided in row (c). Row (b) provides an example of a tool with some degree of wear but well in advance of reaching the end of life criterion. The CWT uses a logarithmic frequency axis, and the time in seconds is along the x axis. The position and colour of each pixel indicates the time, frequency and amplitude of the signals.

As the signal sections are normalized before performing CWT, the magnitude in the transformed images displays the relative intensity of certain frequency bands compared to other sections of the signal. For example, in the final machining pass (c) in Fig. 1 the magnitude of the central band is more significant compared to the others. Thus, showing that the CWTs of bending signals at different stages of the cutting tool life can be distinguished from each other.

For the application to signal prediction, each CWT image generated becomes a frame within a sequence of length 10 frames and each image is converted to a single channel image. The dimensions are reduced to 64 × 64 in order to reduce the computational intensity for training. Any signal below 90 frames was padded evenly on either side with non-machining signal and were split into sequences of 10 frames. The input to this form of network is represented as a 5D tensor, thus the input lengths of the signals must be equal.

For each experiment with a cutting tool, between 8 and 15 passes were reached. In total, 621 sequences of 10 frames were obtained from all machining data and used for training and validation.

The data is processed and filtered using Matlab, and implementations of the models are in Python with Tensorflow/Keras.

3.3. ConvLSTM structure

A total of 12 ConvLSTM networks were developed and tested. Comparisons were made between different ConvLSTM structures with varying number of stacked ConvLSTM layers and kernel sizes.

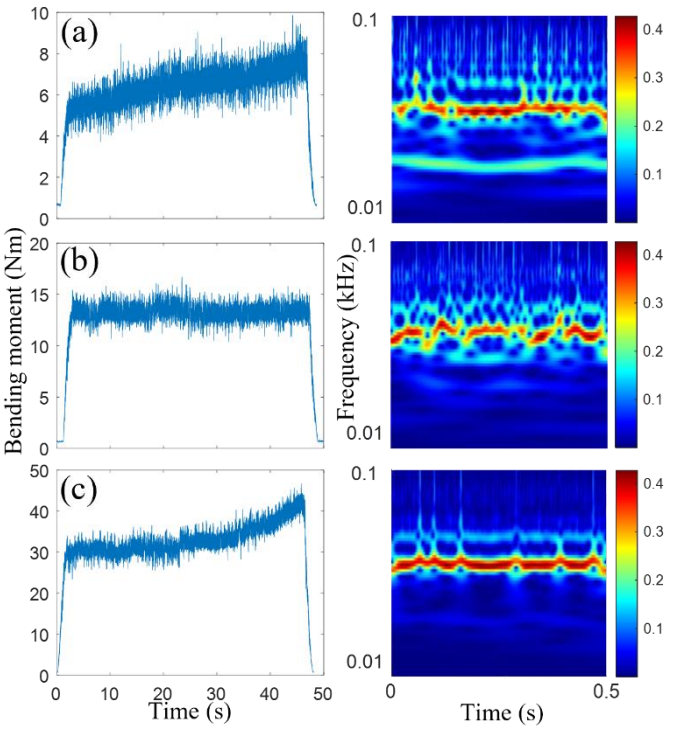

Fig. 1. Plots of the absolute bending moment signal and corresponding CWT images for machining Inconel 718 with (a) new, (b) worn and (c) near end of life cutting tools.

The input dataset was split into 80% training and 20% validation. In order to predict the next frame, the dataset x (originally sequence of length n), will be reduced to frames 1 to $n-1$, and will be shifted by one to create dataset y, which will be frames 2 to n. So that an input frame x_n will be predicting frame y_{n+1}.

As mentioned in section 2, the ConvLSTM network consists of multiple ConvLSTM layers followed by batch normalisation layers sandwiched together in different amounts. Each layer has varying parameters within followed by a 2D convolutional layer. Each ConvLSTM layer had *tanh* activation and hard sigmoid recurrent activation. Binary cross-entropy was used as the loss function, with the 'Adam' optimiser. To reduce computational requirements in the purpose of comparing these combinations, a maximum of four ConvLSTM layers were used in each model.

The kernel size specifies the height and width of the convolutional window in the layer and can determine how much information is captured. One too large will just result in the system becoming fully connected, and too small and it cannot capture all information [18]. The different kernel sizes tested were (3,3), (5,5) and (7,7). Different combinations of these kernel height and widths were used in 2-, 3- and 4-layer networks. In total 12 network structures were trained on the dataset. Each network will be referred to by the kernel sizes, e.g. a network with 3 ConvLSTM layer kernel sizes of (3,3), (5,5) and (7,7) will be referred to as 3-5-7.

Each ConvLSTM network was trained for at most 100 epochs with an input batch size of 5 sequences, chosen to reduce computational resources. For a few models, based on the plots of the training and validation loss at each epoch, there was a point where the validation loss overlapped and became

larger than the training loss, implying that the network is overfitting to this dataset. Techniques such as dropout layers, of which a value of 0.4 was used, were implemented in the ConvLSTM layers to attempt to counteract overfitting.

4. Results

After training, the networks performance in sequence prediction can be tested visually and with comparison metrics to determine how close the sequence prediction is to the ground truth. As the sequence input length is 10; for a random sequence from the validation dataset, the first 5 frames were used to predict the following 5 frames. This is a rolling prediction and from predicted frame 6, frame 7 will be predicted, and so on. An example sequence prediction is shown in Fig. 2 and Fig. 3. These figures display a 10 frame input sequence, where frames 6-10 are predicted based on the previous frame in the sequence.

There are different metrics that can be used to determine image similarity between the frames predicted and the ground truth frames. Each metric mentioned in this section has been applied to the predicted frames and the corresponding ground truth frames for 10 sets of validation sequences and mean values taken; from this, an idea of the consistency for the frame prediction for each model can be shown.

Standard metrics are used to compare the images pixel by pixel, including pixel-wise root mean squared error (RMSE), which is one of the most commonly used metrics for comparison [19].

The average pixel-wise RMSE for the five frames predicted by the six best performing models compared to the actual frames was calculated as shown in Fig. 4. This demonstrated that error increases as the time distance between the current signal frame and the predicted frame increases. With the moving window as the signals are generated during machining, the quality of the prediction increases.

Alongside this standard method, the structural similarity (SSIM) index was used for comparison, which is based on the human vision system and assesses the images based on luminance, contrast, and structural similarity [19]. As shown in Fig. 5, the SSIM indicates a maximum of 65% similarity between the predicted signal for the first frame which drops to 45% for the fifth frame. This is comparable to the results from the pixel-wise mean squared error.

5. Discussion

Visually, from sequence predictions such as that in Fig. 2 and Fig. 3, the ConvLSTM can predict some of the main features within the CWT images. However, there are improvements that can be made, particularly following the first two frame predictions. It is clear that the predicted sequences suffer from low resolution, which increases for each following prediction. This is due to the fact that future predictions are based on frames that have been generated by the model and hence, the prediction quality deteriorates as the time distance between the frames increases. However, in practice with real time data, the prediction accuracy is expected to increase as the modelling progresses.

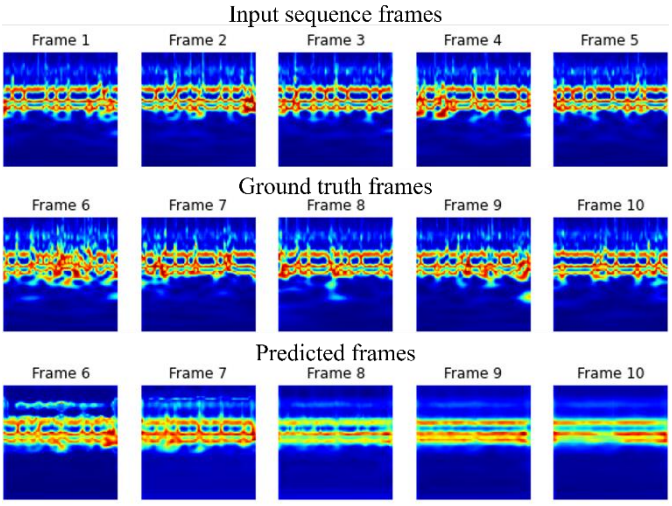

Fig. 2. Example sequence prediction for ConvLSTM network 5-3-3 for machining experiment using a tool with 30° helix and 12° rake angles.

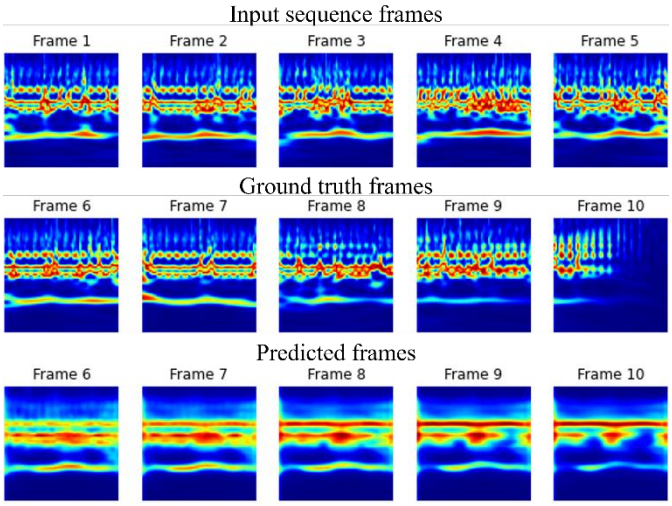

Fig. 3. Example sequence prediction for ConvLSTM network 3-3-3-3 for machining experiment using a tool with 30° helix and 4° rake angles.

The sequence shown in Fig. 3 does display the correct locations and an increase in the magnitude in the main frequency bands, but it highlights the limitation in the frame prediction and struggles to predict the signals at the end of the machining pass. Potentially, this can be overcome by increasing the number of image sequences from the current 5 and training data at the expense of computational power.

The ConvLSTM proposed in this study has shown potential in identifying the main frequencies and features present and their formation across each time period. However, it lacks the ability to determine the correct magnitude of the signals. The accuracy of the predictions tapers off after the second frame prediction and as it gets farther away from the current frame. As the input sequences are only single channel images (initially greyscale), increasing to three channels will present different results. The different frequency bands and patterns in the CWT will be more prominent in different channels and the convolutional operation within the ConvLSTM layers would have a greater chance of being able to extract these bands and channels. However, this will provide more variables for computation and increases the complexity of the model.

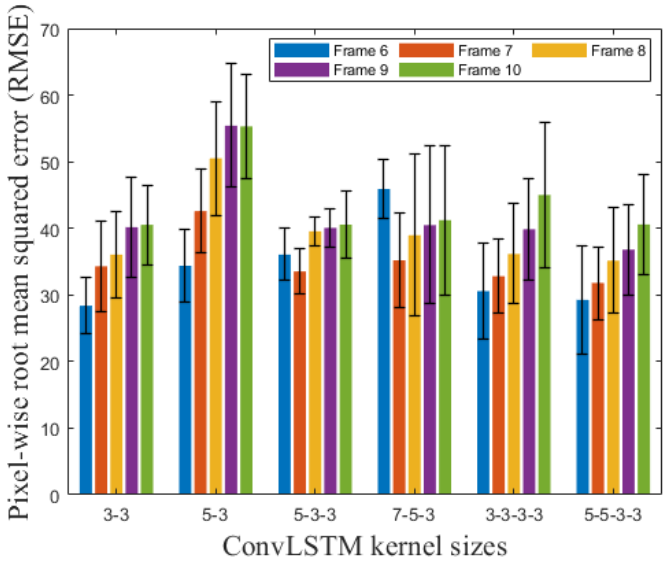

Fig. 4. Average pixel-wise RMSE for six different network structures over ten sample sequence predictions from the validation dataset. Pixel values for each image were in the range of 0 to 255.

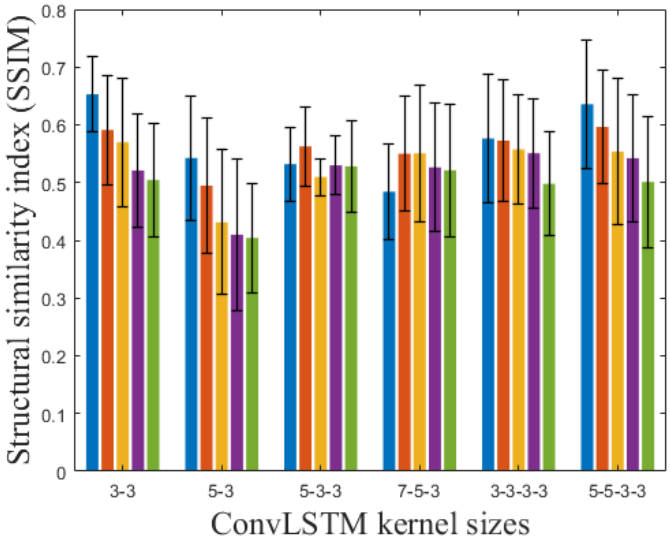

Fig. 5. SSIM for six different network structures.

A lower RMSE and higher SSIM indicate better prediction performances. Comparing the RMSE between the predicted and ground truth frames for the different model structures, the first frame predicted has the most consistently low error between the different models. Interestingly for the 3-layer networks the second frame prediction has a smaller error. Based on this metric, models 3-3 and 5-5-3-3 have the most consistently low pixel-wise RMSE across all frames for sequences predicted. Model 5-3-3 however had the lowest standard deviation for each frame and a moderately low value for every frame, implying that this model has a greater consistency in the repeatability of the frame prediction. There was also smaller variation between the metric values for the frames, showing less resolution being lost in frame predictions further in the future.

Since, RMSE is an average value over the entire image, it gives a general difference between the ground truth and prediction frames. However, it can easily miss similarities in finer details. This is evident when comparing results in Fig. 3 where the overall RSME for this model is lower than that in Fig. 4. but the predictions appear worse visually.

Alongside the six best performing models shown, other combinations of kernel sizes in the ConvLSTM layers were tested. Those that did not finish with a layer with a (3,3) kernel and exclusively used (7,7) or (5,5) kernel sizes performed very poorly in comparison. As the input images are of dimensions 64 × 64, using large kernel sizes means that information is easily lost. The main frequency bands in the CWT images only correspond to at most 5 pixels high. Smaller kernel sizes are needed to pick up these features. If the input dimensions are to be increased, the kernel sizes will have to be scaled. Previous research on the combinations of kernel sizes in layers highlights that the use of multiple kernel sizes is advantageous and kernels must be of a suitable size to capture the ‘movement’ of the features [18].

Some overfitting due to the dataset size was detected as the validation loss started to increase above the training loss after 50 epochs. The main solution would be to increase the input dataset size, which has been shown to improve the prediction RMSE [20]. This work has demonstrated that despite the experimental dataset containing cutting tools with different tool geometries, the model is capable of extracting and predicting prominent features of the input CWT sequences to a reasonable degree. Thus, there is potential for these trained models to be used with input datasets not necessarily of the same cutting tool geometry.

The analysis indicates the feasibility of using ConvLSTM for predicting sensor signals as transformed time-frequency domain signal frames. The prediction is based on immediate data collected for a very short period of time in the future. This would provide enough time for the processing and control systems to make decisions and prevent potentially expensive damages to the workpiece and the machine tool. The current analyses are based on a small set of data which are compacted into frames of 64 × 64 pixels. Evidently, compacting the signals would mean losing potentially valuable information about the machining condition and further investigation is required to identify and optimize the size of the images to ensure minimum loss whilst preventing issues with computational power and long-term data storage.

To incorporate a tool condition monitoring aspect to this model, it should be extended to include a classification network to determine the status of the cutting tool based on the predictions. Fig. 6 Outlines how the current model can be extended to include tool wear classification by incorporating a CNN directly after the ConvLSTM. Both the ConvLSTM and CNN will need to be pretrained on the same datasets, with the input training data to the CNN being CWT images labelled with the tool wear status. Future frame predictions from the ConvLSTM can then be input into the CNN to provide a tool wear classification label.

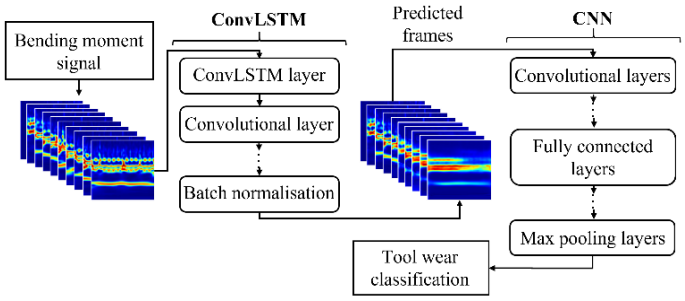

Fig. 6. Proposed ConvLSTM-CNN model

6. Conclusions and future work

Tool bending moment as a function of cutting forces has been identified a direct indicator of cutting tool condition and wear. Being able to predict bending moment allows for tool condition forecasting during machining. A new approach has been proposed in which the time-frequency domain transformation of bending moment signal is treated as a sequence frame and deep learning algorithms are used to predict future frames, e.g. transformed signals. In this paper Convolutional Long Short-Term Memory (ConvLSTM) networks were successfully developed and implemented for sequence prediction of tool bending moment signal data. Different combinations of kernel sizes and ConvLSTM layers were used and the prediction results between these was compared.

The models clearly show the ability to predict future frames of bending moment signals based on an input sequence. The analyses indicate that the kernel size and epochs have direct impact on prediction accuracy. However, there is also a risk of overfitting. Future work will focus on:

- Training the network using a larger dataset and investigate the impact of dimension and depth of signal frames.
- Extending the time distance between the predictions from 100 ms in this study to a few seconds and minutes to obtain longer term predictions for the cutting tool signal.
- Developing and testing a hybrid approach where signals and transformed frames can be used together.
- Assessing the inverse transformation of the predicted CWT frames to the sensor signals.
- Inclusion of a classification algorithm to incorporate cutting tool wear prediction on the predicted frames.

The next steps in this work beyond improving the model and increasing the amount of data is to integrate this algorithm into a system for online tool condition forecasting using a combination of deep learning and physics-based methods. Moreover, the tool bending moment can be correlated with the workpiece geometrical accuracy within a comprehensive machining monitoring system.

Acknowledgements

The authors acknowledge the support of Scorpion Tooling UK Ltd. and the UK Engineering and Physical Sciences Research Council under grant number EP/R513155/1 project reference 2103874.

References

[1] Y. Zhou and W. Xue, “Review of tool condition monitoring methods in milling processes,” *Int. J. Adv. Manuf. Technol.*, vol. 96, no. 5–8, pp. 2509–2523, 2018.
[2] Z. Xie, S. Du, J. Lv, Y. Deng, and S. Jia, “A Hybrid Prognostics Deep Learning Model for Remaining Useful Life Prediction,” *Electronics*, vol. 10, no. 1, p. 39, 2020.
[3] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P. A. Muller, “Deep learning for time series classification: a review,” *Data Min. Knowl. Discov.*, vol. 33, no. 4, pp. 917–963, 2019.
[4] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W. Wong, and W. Woo, “Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting,” *J. Sensors*, vol. 2018, pp. 1–9, 2018.
[5] X. C. Cao, B. Q. Chen, B. Yao, and W. P. He, “Combining translation-invariant wavelet frames and convolutional neural network for intelligent tool wear state identification,” *Comput. Ind.*, vol. 106, pp. 71–84, 2019.
[6] H. Qiao, T. Wang, P. Wang, S. Qiao, and L. Zhang, “A time-distributed spatiotemporal feature learning method for machine health monitoring with multi-sensor time series,” *Sensors (Switzerland)*, vol. 18, no. 9, 2018.
[7] G. Martinez-Arellano, G. Terrazas, and S. Ratchev, “Tool wear classification using time series imaging and deep learning,” *Int. J. Adv. Manuf. Technol.*, pp. 3647–3662, 2019.
[8] X. Cao, B. Chen, B. Yao, and S. Zhuang, “An intelligent milling toolwear monitoring methodology based on convolutional neural network with derived wavelet frames coefficient,” *Appl. Sci.*, vol. 9, no. 18, 2019.
[9] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, “Deep learning for smart manufacturing: Methods and applications,” *J. Manuf. Syst.*, vol. 48, pp. 144–156, 2018.
[10] L. Petruschke, J. Walther, M. Burkhardt, M. Luther, and M. Weigold, “Machine learning based identification of energy states of metal cutting machine tools using load profiles,” *Procedia CIRP*, vol. 104, pp. 357–362, 2021.
[11] F. Hu, J. Qin, Y. Li, Y. Liu, and X. Sun, “Deep Fusion for Energy Consumption Prediction in Additive Manufacturing,” *Procedia CIRP*, vol. 104, pp. 1878–1883, 2021.
[12] P. Stavropoulos, K. Sabatakakis, A. Papacharalampopoulos, and D. Mourtzis, “Infrared (IR) quality assessment of robotized resistance spot welding based on machine learning,” *Int. J. Adv. Manuf. Technol.*, no. 0123456789, 2021.
[13] A. Papacharalampopoulos, C. K. Michail, and P. Stavropoulos, “Manufacturing resilience and agility through processes digital twin: Design and testing applied in the LPBF case,” *Procedia CIRP*, vol. 103, pp. 164–169, 2021.
[14] P. Stavropoulos, A. Papacharalampopoulos, and T. Souflas, “Indirect online tool wear monitoring and model-based identification of process-related signal,” *Adv. Mech. Eng.*, vol. 12, no. 5, pp. 1–12, 2020.
[15] C. Uyulan, “Development of LSTM&CNN based hybrid deep learning model to classify motor imagery tasks,” *Commun. Math. Biol. Neurosci.*, vol. 2021, pp. 1–26, 2021.
[16] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[17] A. Shokrani and J. Betts, “A new hybrid minimum quantity lubrication system for machining difficult-to-cut materials,” *CIRP Ann.*, vol. 69, no. 1, pp. 73–76, 2020.
[18] S. Agethen and W. H. Hsu, “Deep Multi-Kernel Convolutional LSTM Networks and an Attention-Based Mechanism for Videos,” *IEEE Trans. Multimed.*, vol. 22, no. 3, pp. 819–829, 2020.
[19] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.
[20] Z. Yao, Y. Wang, M. Long, and J. Wang, “Unsupervised transfer learning for spatiotemporal predictive networks,” *37th Int. Conf. Mach. Learn. ICML 2020*, vol. PartF16814, pp. 10709–10719, 2020.