Vol.2020-HCI-190 No.11
Vol.2020-UBI-68 No.11
2020/12/9

Preliminary investigation on activity recognition for packaging tasks using motif-guided attention networks

Jaime Morales[1] Naoya Yoshimura[1] Qingxin Xia[1] Takuya Maekawa[1] Atsushi Wada[2] Yasuo Namioka[2]

Abstract: This study presents a method for recognizing packaging tasks using wrist-worn accelerometer sensors under real conditions. As the lead times and actions of packaging activities depend on the number of objects to pack along with the size and shape of each object, it is difficult to recognize operations during every period. We propose a segmentation neural network augmented with a multi-head attention mechanism to capture actions found in a specific operation, which can be useful to identify individual operations. To efficiently detect useful actions with limited training data, we propose an attention guiding approach based on existing motif detection algorithms, which find actions (motifs) that frequently appear in a specific operation. We then use the occurrence of these motifs as a target for each attention head, enabling it to increase its ability to recognize similar operations during the packaging process. We evaluate our framework using data obtained in an actual logistics center.

Keywords: Human activity recognition, Attention Mechanism, Motif identification, Packaging work

1. Introduction

With the increasing availability of wearable sensing devices such as smart-bands, smartwatches, and so forth, the wearable computing research community has actively studied their applications related to human activity recognition in many different fields. Body-worn sensor data for human activity recognition (HAR) can be applied in a variety of applications in both home settings, e.g., daily routine or living conditions for rehabilitation patients, and industrial settings, e.g., process monitoring during assembly work[1], [2], [3], [4], [5]. Our study focuses on the industrial application, specifically on the identification of operations in the packaging process at a logistics center using data from wrist-worn acceleration sensors.

While the existence of logistics centers has been a part of all product chains for a long time, given the proliferation of delivery services such as Amazon or Alibaba, logistics centers have become one of the more rapidly expanding industries worldwide. Furthermore, the generality of products that go through logistics centers has equally increased, allowing a single center to be used for packaging hundreds or thousands of different items using the same process.

This extension in the reach of logistics centers puts special importance on the improvement of packaging tasks both from the process structure and worker ability perspectives. Improving packaging efficiency would result in reduced time spent per item and in turn decrease delivery time and costs associated with the packaging process. This paper aims to provide a tool that allows a manager in a logistics center to improve the process structure and increase the efficiency of the packaging work by identifying the order and length of the standardized operations performed during the packaging process.

A packaging work is composed of a sequence of operations that culminates in one or more items inside a container ready to be transported to a different location. We can say that packaging work consists of a set of periods,

[1] Department of Multimedia Engineering, Graduate School of Information Science and Technology, Osaka University
[2] Corporate Manufacturing Engineering Center, Toshiba Corporation

where an item or group of items is packed. While each of these periods includes a standard number of operations the worker should perform, due to the nature of packaging work each period may differ on the length of each particular operation or on the order of operations depending on the size and number of objects each package contains. The variability in the length and order of these operations is reflected in the data captured from the accelerometer sensors which varies as well from one period to the next making it difficult for standard models to recognize similar operations with the limited amount of data collected.

Our approach is based on the fact that even when the order and length of packaging operations are not uniform among periods, each operation has some unique motions that can be used to identify and differentiate operations. First, since our objective is to identify the start and finish of each operation we base our method on a segmentation neural network, i.e., U-Net [1]. We use this base network to better segment each specific operation with a dense prediction function. Then, in order to identify the unique motions that describe each standard operation, we augment the U-Net with a multi-head attention mechanism. However, this mechanism would originally require a huge amount of data which is hard to acquire under real conditions in logistic centers so we employ existing motif finding algorithms to create a target occurrence sequence for known unique motions to aid the attention mechanism training. We call our method motif-guided-attention network (MGA-Net).

The contributions of this study are summarized as follows.

- □ Our method improves the activity recognition accuracy with an attention guiding approach on a limited data set from packaging logistics works.
- □ We propose a motif guiding method for attention mechanisms to improve its accuracy on small training data sets.
- □ We propose three kinds of motifs corresponding to characteristic actions that can enhance the segmentation.
- □ We evaluate the proposed method using sensor data collected from an actual logistics center. Using this data, the proposed method outperformed baseline methods based on recurrent neural networks and segmentation networks.

2. Related Work

Due to the recent growing interests in smart manufacturing, more studies focus on recognizing and supporting factory activities using variety of sensors. In particular, acceleration sensor which has a good trade-off between activity prediction accuracy and power consumption has been widely used in ubicomp community. For example, Maekawa et al. [4] reused labeled sensor data of source users who have similar physical traits to a target user to train the target user's activity model. State-of-the-art convolutional neural networks (CNN) have also been applied on sensor-based activity recognition. Rueda et al. [6] applied parallel branches within a CNN, with a branch processing data from each inertial sensor, confirming that the proposed network outperformed a baseline CNN when using a small amount of sensor data in the logistics domain. Zhang et al. [7] indicated that a U-Net based algorithm is able to support both activity labelling and prediction at each sensor data point. Cordonnier et al. [8] and Murahari et al. [9] utilized different types of attention mechanisms to generate higher-dimensional feature representations used for activity recognition.

Several studies explored motif finding algorithms for activity and gesture recognition. Minnen et al. [10] first discovered motif seeds using a minimum description length criterion and then refined the motif seed by splitting, merging, and extending the motifs. Berlin et al. [11] recognized leisure activities by employing useful motifs in acceleration signals. Maekawa et al. [12] measured the duration of each work period on a production line in an unsupervised manner, by discovering a motif that appears only once in each work period. Xia et al. [13] further proposed two types of motifs, which correspond to specific actions that appear once or several times in a work period to robustly recognize operations even outliers exist.

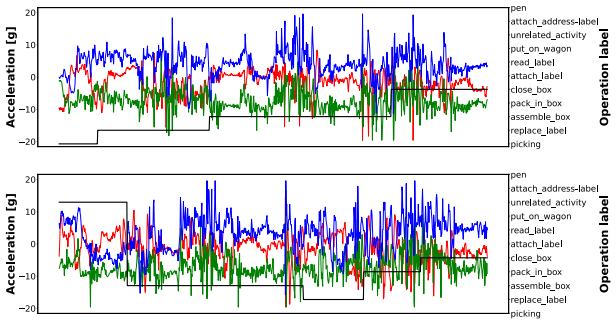

図 1: An introduction to logistic work accelerometer data.

3. Activity Recognition Method

3.1 Preliminaries

We assume a data set captured from workers doing packaging work in a logistics center. The data are tri-axial accelerometer data captured from the workers using a commercial smartwatch on their dominant hand. A worker iterates a period of work consisting of performing sequential operations from picking a product, assembling a box, attaching a label, closing the box, and finally placing the finished box on a cart.

In Figure 1 we can see example data collected from a worker's watch, the data on the top corresponds to the first period and the operations of replace-label, assemble-box, and close-box, the data on the bottom corresponds to the same worker performing the same operations during the 15th period. Even when the worker performs the same operation for both periods, the data corresponding to each of them is not uniform in shape or length. In this case, we can also see how the worker has inverted the order of two of the operations. Our study aims to classify each of these data points into an activity class that corresponds to the operation the worker performs, allowing us to know the starting and ending time of each operation.

3.2 Method overview

Our method recognizes the operations performed during the packaging task using a segmentation neural network. We use this network since the operations are performed sequentially but with different lengths. Based on U-Net [7], we propose an improvement over this segmentation network idea by augmenting the network structure adding a multi-head attention mechanism. This new layer helps identify important actions in an operation. Unfortunately, training such a complex network is data-hungry, which is not suitable for a logistics environment, where preparing enough training data is difficult. Therefore, we utilize existing motif detection algorithms [13] to find useful actions/motions that can define the operations we are looking to recognize. Such actions may be unique to a particular operation like placing the tape on the box when closing it, or removing the glue protection before attaching a label. We use such motifs to guide the training of the modified network to make up for the amount of training data available.

In Figure 2 we can see an overview of our method, initially, we have the accelerometer data time-series which

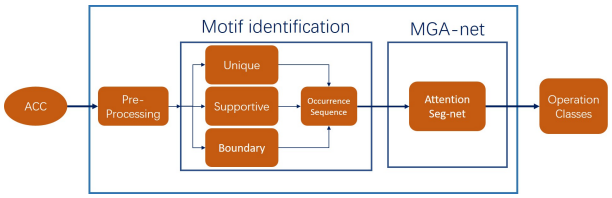

図 2: Method Overview.

is then pre-processed inside our motif finding algorithm. Once pre-processed we employ three different techniques to obtain a **Unique**, **Supporting**, and **Boundary motifs**. These motifs can be understood as follows:

- The unique motif corresponds to an action that occurs almost solely on the expected operation.
- The supportive motif is an action closely related to the unique motif appearing almost always at the same distance from it in time.
- The boundary motif could be an action that is performed at the start or finish of the operation.

We then calculate a Motif Occurrence Sequence that is in turn used to train the attention augmented segmentation network. The network is a motif guided attention network (MGA-Net), it performs activity recognition on small training sets using a segmentation network with a guided attention mechanism based on the found motifs.

3.3 Motif Selection

3.3.1 Data preprocessing

In order to reduce computational costs, we first apply principal components analysis (PCA) to simplify the input acceleration data into one-dimension. Then, we use piecewise aggregate approximation (PAA) to reduce the number of data points of the sequence. After that, we symbolize the sequence based on previous factory activity recognition methods [12]. In brief, we convert each of the aggregate values in the sequence into a symbol based on the thresholds of value range (e.g., a sequence is symbolized to *aabcddbaa*, where the same character belongs to the same value range). We then use this symbolized sequence to track motifs.

3.3.2 Occurence sequence calculation

We start by extracting all possible motifs from the initial working period. Given that we do not know the position or frequency of characteristic actions for each operation, we perform an initial scoring of the candidates to reduce the computation time. To perform said pruning, we isolate all sub-sequences that correspond to that operation and scan for more occurrences of the candidate motifs while calculating the number of appearances it has

among all periods. From this, we select the best group of candidate motifs for selection.

Now that we have our group of candidate motifs for each operation, we slide each candidate motif along the complete training time-series calculating its similarity ratio to generate an occurrence sequence. After extracting a group of candidate motifs for each operation, we compare the similarity of each candidate motif sliding along the complete sequence to get an occurrence sequence. To obtain the similarity ratio we employ the Levenshtein distance metric [14].

$$L_{ratio} = (lensum - ldist)/lensum \tag{1}$$

The ratio is given by Equation 1 where *lensum* is the length of the maximum sequence and *ldist* is the Levenshtein cost for symbol modification between the sequences.

We then process the sequences in order to transform the similarity sequence into an occurrence sequence. Since we only want the position where the action occurs we eliminate all sequence values below the threshold of 0.9 similarity ratio. After these calculations, we are left with several final candidate sequences that may be helpful to recognize each of our classes.

3.3.3 Unique Motif selection

A unique motif that usually appears in a specific operation is useful to identify that operation. However, from our pool of candidate motifs, some of them may correspond to motions that are not unique to one operation but several. In order to identify which of our candidate motifs best describes our operations, we devised a scoring system based on correct and incorrect appearances. To calculate the uniqueness score of a motif that corresponds to an operation O_j we first generate a supporting sequence **OV** which corresponds to the value given to the operation so that when timestep t is inside O_j the value of $\mathbf{OV}_t$ is positive and negative otherwise. With this, we calculate the uniqueness score of the motif with Equation 3. Where t corresponds to the current timestep and T is the total length of the sequence.

$$\mathbf{OV}_t = \begin{cases} 1, & \text{if } t \in O_j \\ -1, & \text{otherwise} \end{cases} \tag{2}$$

$$U_{score} = \sum_{t=0}^{T} L_{ratio_t} \mathbf{OV}_t \tag{3}$$

Using this calculation we select the motif that mostly appears in the operation O_j. We call this motif the Unique Motif and there is one associated with each of our classes or operations.

3.3.4 Supportive Motif selection

As an operation consists of a sequence of actions, the corresponding unique motif can only locate a single action/motion but it is still difficult to review the structure of the operation. To efficiently recognize the operation, capturing the sequential structure of actions is important. Therefore, we leverage a supportive motif for the unique motif that occurs before or after the unique motif, where the time difference is consistent with the corresponding unique motif (e.g., a supportive motif can be removing the glue cover before pasting a label on the box. This action always happens when a worker attaches a label.). To find the supportive motif we first identify all periods P where the candidate appears along with the Unique Motif in the same period. Once we have identified all these periods we calculate a distance-vector **MD** that contains the symbol distance from the Unique Motif. We then obtain the supportive score (S_{score}) for each motif by calculating the average standard deviation inside **MD** and multiplying the result by $\frac{P}{2}$ so giving a higher score to candidates with more occurrences together along with Unique Motif.

$$S_{score} = \sqrt{\frac{(\sum_{i=0}^{P}(\mathbf{MD}_i - \overline{\mathbf{MD}}))}{P-1}}\frac{P}{2} \tag{4}$$

We then select the candidate with the highest S_{score} as Supportive Motif. The occurrence of both Motifs during the same period gives higher confidence when identifying an operation compared to only using a single Unique Motif.

3.3.5 Boundary Motif selection

Different from the supportive motif, a boundary motif always appears at the beginning or end of an operation. Since we employ a segmentation neural network to find the beginning and ending times of each operation, it is important to know where the boundary between operations occurs. Unlike the scoring systems used to select the Unique and Supportive motifs, the boundary motif is selected by scanning the sequence with the remaining candidate motifs and counting the number of appearances for each of them within a specified distance from the end or the start of their designated operation. However, a secondary count is also performed to ensure this Boundary motif appears along with the other 2 selected motifs. As shown in Equation 5, we score the boundary motifs by simply multiplying the number of appearances close to the boundary (BA) by the number of appearances it has together with the other motifs, where each occurrence of

all three motifs appearing on the same period is aggregated to the value of TA. We then select as boundary motif the candidate with the highest B_{score}.

$$B_{score} = BA * TA \tag{5}$$

3.4 MGA-Net

3.4.1 Network Structure

As noted in section 3.2, we base our network structure on the existing U-Net topology for segmentation of time-series data. We can see in Figure 3 our network structure which consists of three encoding blocks as well as three decoding blocks. The topology for a single block consists of 2-one dimensional convolutional layers plus one either max-pooling layer or upward convolutional layer depending on if the block is part of the decoder or encoder. Another part of the original topology is the concatenation layer at the beginning of each new decoder block where the output from the second convolutional layer of the corresponding block is concatenated with the output of the up-convolutional layer. This network structure allows us to use a complete period of data as a single input. It is also capable of giving a classification to every independent data point in the sequence.

As seen in Figure 3, we include at the end of the first encoding block a multi-head attention mechanism whose purpose becomes finding specific actions that are useful for recognizing individual operations from one another. This mechanism consists of one attention head dedicated to identifying the specific actions that correspond to one of our classes, that way each head analyses the time-series independently and can focus better on finding the required data. We introduce this attention mechanism on the first layer for it to be able to focus on even small actions/motions that would be lost on deeper layers. We also chose this position since the attention will in turn be concatenated with the final decoding block carrying double importance in the network.

3.4.2 Network Training

Another modification we made from the previous introductions of U-Net for activity recognition is the way we train the network. Network training in this network is usually done to reduce segmentation error but fails to overcome the lack of descriptive features to recognize specific operations. Furthermore, given our lack of training data, the accuracy of a self-attention layer does little to improve the recognition results. For this, we introduce a new method to train the network using the motifs found

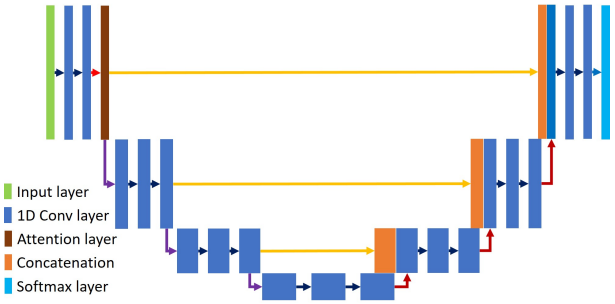

図 3: MGA-Net Network structure.

in section 3.3.

For each operation in our group of classes, we have prepared a total of three occurrence sequences corresponding to the Unique, Supportive, and Boundary motifs. Those sequences point out where key actions occur for that operation and can therefore be a meaningful help to train the attention mechanism for it to find similarly meaningful data along the time-series. Before using these sequences to train, they must be simplified. For this, we add the three sequences into a single Motif Occurrence Sequence. We further normalize the distribution of said sequence to assimilate as a Softmax distribution and equalling the sum of its values to 1.

Now for the training procedure, we must first analyze how the U-Net is trained. Each layer l on the network has an output vector that may be denoted as:

$$\mathbf{z_i} = f(\{x_{i+i'}\}_{-\frac{k_i-1}{2} \leq i \leq \frac{k_i-1}{2}}) \tag{6}$$

Where $\mathbf{z_i}$ corresponds to the intermediate output vector of said layer, k_i corresponds to the kernel size of the i^{th} layer, and $f(.)$ denotes the activation function performed inside the layer. The size of the vector can be expressed as $1 \times w_{l+1} \times f_{l+1}$ where w_{l+1} is the length of the output vector and f_{l+1} is the number of feature maps after the operation. We can then obtain the loss function for this same layer l denoted as:

$$\mathbf{L}(x, z_i; W, b) = \sum_{j}^{N} -\log p(z_{i_j}|x, W, b) \tag{7}$$

$$\mathbf{L}(x, z_i, \mathbf{MOS}; W, b) = \sum_{j}^{T} -\log p(z_{i_j}|x, W, b) + \lambda(z_{i_j} - \mathbf{MOS}_j) \tag{8}$$

Where N is the total length of the output vector and z_{i_j} is a single sample in the output sequence. In the case of the attention layer, the output vector is of size z_i is $1 \times T \times AH$ where T is the total length of the time-series and AH is the number of attention heads in the layer.

We then introduce a new term for the motif guided training given to the attention layer. The motif occurrence sequence is introduced as a parameter for loss calculation. The attention loss is now calculated with respect to both, the expected output class and the value of similarity obtained from the Motif Occurrence Sequence (MOS). With this new term, we update Equation 7 and obtain the updated loss function for the attention layer as presented on Equation 8. Where $\mathbf{MOS}_j$ corresponds to the subset of values for the j^{th} timestep from the combined MOS of all existing operations. As a final note, we introduce a stabilizing parameter λ given that during training the loss value of the standard segmentation loss overshadows the attention training loss introduced. Said parameter can be adjusted to increase or decrease the effect of the motif guided attention loss training in the network. Given that we train this network for continuous cycles we require an optimization function to update the weight parameters of the various layers in the model during back-propagation. For simplicity, we have decided to include all layers under the same optimizer and use an Adam optimizer for this purpose.

Worker	Periods	Number of samples	Values per sample	Classes in data set
1	46	148076	3 (x,y,z)	8
2	28	308207	3 (x,y,z)	9
3	31	232507	3 (x,y,z)	9
4	73	375508	3 (x,y,z)	10

表 1: Overview of recorded data sets.

4. Evaluation

4.1 Dataset

We evaluated the proposed method using 4 data sets collected from 4 individuals working in a real logistics center. Table 1 shows an overview of the data sets. The accelerometer data were collected from a smartwatch (Sony SmartWatch3 SWR50) worn on the worker's dominant side wrist, with an approximate sampling rate of 60Hz. Due to temporary constraints as well as differences in the ability of the worker and nature of the packaging tasks available, the total number of periods recorded for each individual varied. The data was labeled into classes using the name of 10 packaging operations, not all workers perform every operation.

4.2 Evaluation Methodology

For each data set, we use leave one out cross-validation as evaluation criteria among methods. We use the weighted average F1-score as metric to compare the performance of the proposed method against other models.

We provide the results for the proposed method and other comparing methods to evaluate the effectiveness of our approach. The methods to be tested are listed as follows:

- **LSTM**: As initial baseline we use a five-layer Long short term memory (LSTM) network. As input we feed a segment within a sliding time window with a size of 60 points and slide length of 10 points. The training period was of 100 epochs with a batch size of 128.
- **U-Net**: The base of our method corresponds to the segmentation neural network referred to as U-Net. We use a network that consists of 3 encoding blocks and 3 decoding blocks. For each encoder and decoder block, we use two 1D convolutional layers with a kernel size of 3x1 and a max-pooling/up-convolutional layer with a kernel of 2x1. We train this network using the negative logarithmic likelihood loss function as presented on Equation 7 under an Adam optimizer. The training period was 100 epochs with a batch size of 4.
- **Self-attention U-Net**: The Self-attention network is an extension of the previous U-Net architecture, with an added self-attention mechanism at the end of the first encoding level.
- **Proposed (MGA-Net)**: This is our proposed method.

4.3 Results

Figure 4 compares the performance of the 4 methods among all the workers. The proposed method (MGA-Net) achieved the highest average accuracy across all 4 data sets. We can observe how the lack of training data does not allow the self-attention model to really capture the complete importance of the existing characteristic actions present on the operations and only slightly improves in performance when compared to the pure U-Net model. However, by introducing our training mechanism to this same architecture the performance of the network increases on average 47% when compared to the original U-Net. When comparing performance among the workers it is clear that the lack of training data is the biggest concern when dealing with any type of deep learning method-

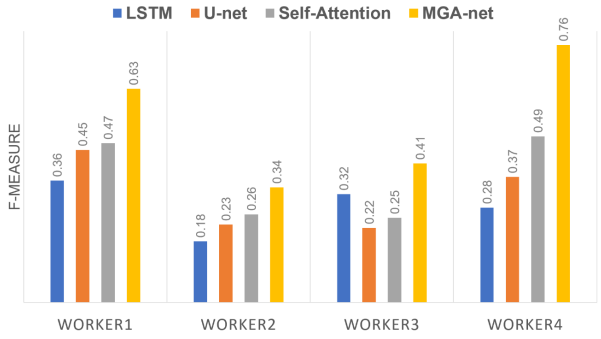

図 4: Accuracies (F-measures) of the 4 methods among all data sets.

ology, nevertheless, we can see while observing worker 2 and worker 3 that our motif guided attention mechanism still outperforms the rest of the architectures.

Figure 5 gives us a graphic perspective of how each model identifies or miss-identifies each particular operation. We chose the matrices belonging to worker 4 since, on this data set, the performance increase of our method is almost threefold when compared to the LSTM and twofold when compared to the U-Net. First, we analyze the stability of all methods and then the reasons that may explain the differing performance among them.

We now focus on Figure 6, where we can observe the performance of the different methods on a single working period. This figure shows us the characteristics of each model. The LSTM model has the worst overall performance since it fails to recognize complete operations. Our second baseline, the U-Net model can be seen trying to gather as much close related points as possible but failing to decide on a single operation as it alternates between different classes. Then we see how the introduction of the attention mechanism provides accuracy in the selection of a single class to the segmentation model. Our method proves its advantage by disappearing behind the ground truth label for almost the complete length of the working period. However, as we can see in the picking segment it completely fails to recognize the operation, this may be possible if that the worker did not perform the activities similar to the actions found during the motif selection.

We use Figure 7 to analyze the functionality of our motif guided attention training. In this figure, we can observe the attention scores corresponding to the attention heads dedicated to identifying the operations of close-box and replace-label, as well as the input acceleration sequences used for this testing period along with the ground truth and predicted labels. We can observe how the trained attention heads dedicated to each operation have the ability to find distinctive motions and then increase the importance of the segments around them.

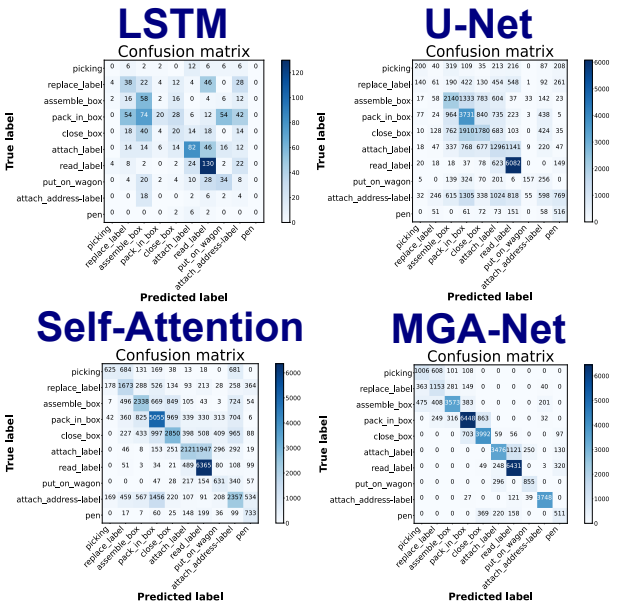

図 5: Confusion matrices for worker 4, operation recognition results among 4 comparison methods with LSTM, U-Net, Self-attention U-Net and MGA-Net.

In this figure, we highlight 3 segments from the time-series where the functionality of our method is best described (red circles). On the first highlighted section we observe particularly small values of attention but an accurate recognition, this shows the impact the rest of the series around those points have when using a segmentation architecture for our model. On the second highlight, there is an unstable recognition even when the attention score is high, which proves that training is not fully directed by the motif. Finally, at the end of the series we see a clear increase of the attention values during the pen operation, since the pen operation refers to a worker writing something by hand, it is highly rare for it to appear during any given period and can be easily mistaken for a more common operation such as closing a box, more so if the worker performs a motion similar to the one you would expect during that operation (e.g. flipping the box).

5. Conclusion

We conclude that the preliminary investigation on the application of motif-finding techniques for attention mechanism training constitutes an advantage when attempting activity recognition with small data sets. Furthermore, we have proven that by using our three motifs extracted from particular motions as a base for training the attention mechanism we could make it increase its ability at

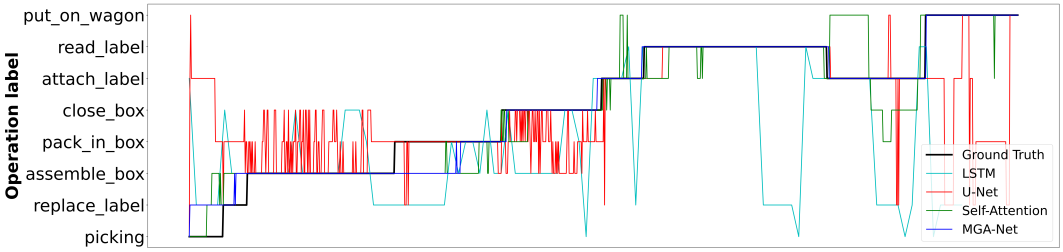

図 6: Prediction results comparison for the 4^{th} period from worker 1.

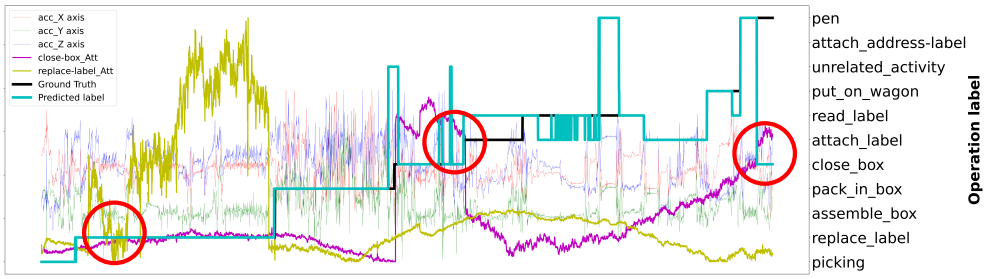

図 7: MGA-Net Attention score of close-box and replace-label operation with predicted and ground truth labels for the 4^{th} period from worker 3.

an improved rate. This optimization may allow the use of the extremely helpful attention mechanism to be applied in more applications where the collection of data proves difficult as is the case of a logistics center.

As future work, we desire to further increase the ability of the motif guided attention technique and MGA-net model to solidify the preliminary advances made in this study.

6. Acknowledgments

This work is partially supported by JST CREST JP-MJCR15E2, JSPS KAKENHI Grant Number JP16H06539 and JP17H04679.

参考文献

[1] Bao, L. and Intille, S. S.: Activity recognition from user-annotated acceleration data, *Pervasive 2004*, pp. 1–17 (2004).

[2] Chavarriaga, R., Sagha, H., Calatroni, A., Digumarti, S. T., Tröster, G., Millán, J. d. R. and Roggen, D.: The Opportunity challenge: A benchmark database for on-body sensor-based activity recognition, *Pattern Recognition Letters*, Vol. 34, No. 15, pp. 2033–2042 (2013).

[3] Korpela, J., Takase, K., Hirashima, T., Maekawa, T., Eberle, J., Chakraborty, D. and Aberer, K.: An energy-aware method for the joint recognition of activities and gestures using wearable sensors, *International Symposium on Wearable Computers (ISWC 2015)*, pp. 101–108 (2015).

[4] Maekawa, T. and Watanabe, S.: Unsupervised activity recognition with user's physical characteristics data, *International Symposium on Wearable Computers (ISWC 2011)*, pp. 89–96 (2011).

[5] Reining, C., Niemann, F., Moya Rueda, F., Fink, G. A. and ten Hompel, M.: Human Activity Recognition for Production and Logistics—A Systematic Literature Review, *Information*, Vol. 10, No. 8, p. 245 (2019).

[6] Moya Rueda, F., Grzeszick, R., Fink, G. A., Feldhorst, S. and Ten Hompel, M.: Convolutional neural networks for human activity recognition using body-worn sensors, *Informatics*, Vol. 5, No. 2, Multidisciplinary Digital Publishing Institute, p. 26 (2018).

[7] Zhang, Y., Zhang, Y., Zhang, Z., Bao, J. and Song, Y.: Human activity recognition based on time series analysis using U-Net, *arXiv preprint arXiv:1809.08113* (2018).

[8] Cordonnier, J.-B., Loukas, A. and Jaggi, M.: On the relationship between self-attention and convolutional layers, *arXiv preprint arXiv:1911.03584* (2019).

[9] Murahari, V. S. and Plötz, T.: On attention models for human activity recognition, *The 2018 ACM International Symposium on Wearable Computers*, pp. 100–103 (2018).

[10] Minnen, D., Starner, T., Essa, I. and Isbell, C.: Discovering characteristic actions from on-body sensor data, *2006 10th IEEE International Symposium on Wearable Computers*, IEEE, pp. 11–18 (2006).

[11] Berlin, E. and Van Laerhoven, K.: Detecting leisure activities with dense motif discovery, *The 2012 ACM Conference on Ubiquitous Computing*, pp. 250–259 (2012).

[12] Maekawa, T., Nakai, D., Ohara, K. and Namioka, Y.: Toward practical factory activity recognition: unsupervised understanding of repetitive assembly work in a factory, *UbiComp 2016*, pp. 1088–1099 (2016).

[13] Xia, Q., Korpela, J., Namioka, Y. and Maekawa, T.: Robust Unsupervised Factory Activity Recognition with Body-worn Accelerometer Using Temporal Structure of Multiple Sensor Data Motifs, *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, Vol. 4, No. 3, pp. 1–30 (2020).

[14] Yujian, L. and Bo, L.: A Normalized Levenshtein Distance Metric, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 6, pp. 1091–1095 (online), DOI: 10.1109/TPAMI.2007.1078 (2007).