Spring 2020

Attention mechanism in deep neural networks for computer vision tasks

Haohan Li

Department: Computer Science

ATTENTION MECHANISM IN DEEP NEURAL NETWORKS FOR COMPUTER VISION TASKS

by

HAOHAN LI

A DISSERTATION

Presented to the Graduate Faculty of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

DOCTOR OF PHILOSOPHY

in

COMPUTER SCIENCE

2020

Approved by

Zhaozheng Yin, Advisor
Jennifer Leopold
Patrick Taylor
Ruwen Qin
Venkata Sriram Siddhardh Nadendla

ABSTRACT

Attention mechanism, which is one of the most important algorithms in the deep Learning community, was initially designed in the natural language processing for enhancing the feature representation of key sentence fragments over the context. In recent years, the attention mechanism has been widely adopted in solving computer vision tasks by guiding deep neural networks (DNNs) to focus on specific image features for better understanding the semantic information of the image. However, the attention mechanism is not only capable of helping DNNs understand semantics, but also useful for the feature fusion, visual cue discovering, and temporal information selection, which are seldom researched. In this study, we take the classic attention mechanism a step further by proposing the *Semantic Attention Guidance Unit* (SAGU) for multi-level feature fusion to tackle the challenging Biomedical Image Segmentation task. Furthermore, we propose a novel framework that consists of (1) *Semantic Attention Unit* (SAU), which is an advanced version of SAGU for adaptively bringing high-level semantics to mid-level features, (2) *Two-level Spatial Attention Module* (TSPAM) for discovering multiple visual cues within the image, and (3) *Temporal Attention Module* (TAM) for temporal information selection to solve the Video-based Person Re-identification task. To validate our newly proposed attention mechanisms, extensive experiments are conducted on challenging datasets. Our methods obtain competitive performance and outperform state-of-the-art methods. Selective publications are also presented in the Appendix.

ACKNOWLEDGMENTS

I would like to sincerely thank my advisor, Dr. Zhaozheng Yin, for his selfless help and support to my research. I am also grateful to the members of my committee for their patience and support in overcoming numerous obstacles I have been facing through my research and defense.

TABLE OF CONTENTS

Page

APPENDICES

LIST OF ILLUSTRATIONS

Figure Page

LIST OF TABLES

1. INTRODUCTION

Attention Mechanism (AM), which is firstly introduced for Natual Language Processing (NPL) Bahdanau *et al.* (2014), has now become enormously popular in the computer vision community as an essential embeddable component of Deep Neural Networks (DNN) for solving different tasks, such as image classification, semantic segmentation, and video analysis.

The design intuition of attention mechanism can be explained using the human optical processing system, which tends to focus on particular parts of the image while ignoring other irrelevant information in a manner that can assist in the perception Xu *et al.* (2015). Similarly, the major role of traditional AM in DNNs is to help the DNN to focus on the specific features of the input data to learn the semantic meaning for improving the network performance effectively and efficiently, instead of letting the DNN indiscriminately learn every available feature.

Despite the notable performance improvements achieved by using AM to assist DNN in different computer vision tasks, AM is not just limited in helping DNNs learning semantic information. In this study, we first take the traditional attention mechanism a step further in Section 2, by introducing the Semantic Attention Guidance Unit (SAGU) that can exploit the high-level semantic information as soft self-attentions that lead low- and mid-level features to focus on target areas and highlight the feature activations that are relevant to the target instance. In Section 2, we embed our SAGU in a newly designed Fully Convolutional Network that can be combined with an annotation suggestion algorithm to assemble a deep active learning framework, which can solve the challenging biomedical image segmentation task with small amount of annotated data for training. In Section 3, we further introduce our Two-level Spatial and Temporal Attention Network (TSTAN), which consists of three major components: (1) Semantic Attention Unit, which is an updated version of our SAGU, can capture more flexible and discriminative correlations between mid- and high-level features; (2) Two-level Spatial Attention Module that can discover multiple different visual cues within one image; (3) Temporal Attention Module that can assign multiple temporal attention weights to each frame of the video to represent the feature importances of the discovered visual cues. The proposed TSTAN is capable of handling the notorious temporal information inconsistency challenge in video-based person re-identification. In Section 4, we conclude our newly proposed attention mechanisms in the two challenging tasks and introduce our future work on improving the accuracy of attention mechanism.

2. ATTENTION, SUGGESTION AND ANNOTATION: A DEEP ACTIVE LEARNING FRAMEWORK FOR BIOMEDICAL IMAGE SEGMENTATION

2.1. RESEARCH BACKGROUND

Automated image segmentation is a cornerstone of many image analysis applications. Recently, thanks to their representation power and generalization capability, deep learning models have achieved superior performance in many image segmentation tasks Liao *et al.* (2017) Kamnitsas *et al.* (2017) Oda *et al.* (2017). However, despite the success, deep learning based segmentation methods still face a critical hindrance: the difficulty in acquiring sufficient training data due to the high annotation cost. In biomedical image segmentation practices, this hindrance can be more challenging for the reason that: (1) Only domain experts can provide precise annotations for biomedical image segmentation tasks, which makes crowd-computing quite difficult; (2) Biomedical images from high throughput experiments contain much more instances than natural scene images, which requires extensive workforces to provide pixel-level annotations; (3) Due to the dramatic variations in biomedical images (e,g, different modalities, image settings, shapes and appearances of the target objects, etc.), deep learning models need a specific set of training data for each segmentation task to achieve competitive performances, rather than using a general training dataset to solve all kinds of segmentation tasks. Therefore, given a new segmentation task with a large unlabeled training dataset, how to select representative data for annotation is important to achieve competitive performance with less human efforts.

2.1.1. Related Works. To alleviate the burden of manual annotation in biomedical image segmentation tasks, several attempts have been made in recent years. An array of weakly supervised segmentation algorithms Papandreou *et al.* (2015) Xiao *et al.* (2018) Hong *et al.* has been proposed. However, how to select representative data samples for annotation is overlooked. To address this problem, active learning can be utilized as an annotation suggestion process to query informative samples for annotation for the high-quality performance Settles (2009) Zhang (2011). As shown in Dutt Jain and Grauman (2016), using active learning, good performance can be achieved using significantly less training data in natural scene image segmentation. However, this method is based on the pre-trained region proposal model and pre-trained image descriptor network, which cannot be easily acquired in the biomedical imaging field due to the large variations in various biomedical applications. A progressively trained active learning framework is proposed in Yang *et al.* (2017). However, the framework only focuses on the uncertainty and the representativeness of suggested samples in the unlabeled set and ignores the rarity of suggested samples in the labeled set, which can easily incur serious redundancy in the labeled set.

2.1.2. Our Proposal and Contribution. To significantly alleviate the burden of manual labeling, we propose a deep active learning framework combining the deep learning model with attention mechanism and the active learning algorithm, which iteratively suggests the most valuable annotation samples to improve the model's segmentation performance progressively. Instead of using pre-trained image descriptor networks that require extra training efforts, we exploit the deep learning model in the proposed framework to obtain domain-specific image descriptor and directly generate segmentation. To address the redundancy issue in the labeled set, we design an active learning algorithm that considers not only the uncertainty and representativeness of the suggested samples in the unlabeled set but also the rarity of the suggested samples in the labeled set.

Although the above proposal seems to be straightforward, it is challenging to design a framework that can perfectly integrate the deep learning model into an active learning process due to the following challenges: (1) The deep learning model needs to be of good generalization capability so that it can produce reasonable results when little training data is available in the active learning process; (2) The deep learning model needs to perform well when using the entire training set so that it can provide a good upper bound for

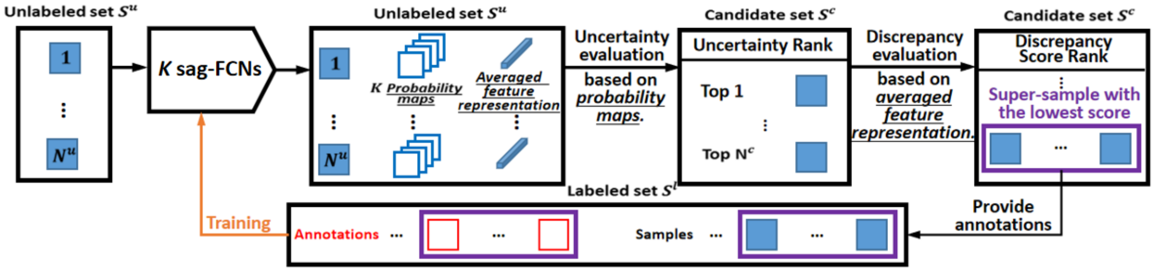


Figure 2.1. The workflow of our deep active learning framework.

the active learning framework; (3) The active learning algorithm should be capable of making judicious annotation suggestions based on the limited information provided by a not-well-trained deep learning model in the early training stage.

To overcome these three challenges, we design a deep active learning framework that consists of two major components, the semantic attention guided Fully Convolutional Networks (sag-FCN) and the distribution discrepancy based active learning algorithm (dd-AL) :

- Attention: For the first and second challenges, we carefully design the sag-FCN that uses semantic attention guidance units (SAGUs) to automatically highlight salient features of the target content for accurate pixel-wise predictions. In addition, both of the sag-FCN and the SAGU are built using bottleneck designs to significantly reduce the number of parameters while maintaining the same number of feature channels at the end of each residual module. This design ensures the good generality of the proposed sag-FCN.

- Suggestion and Annotation: For the third challenge, we design the dd-AL that reveals the final goal of the iterative annotation suggestion process: decreasing the distribution discrepancy between the labeled set and the unlabeled set (note, in this paper, the labeled set and unlabeled set refer to the labeled and unlabeled portions of a training dataset, respectively). If the discrepancy between these two sets are small enough, which also means their distribution is similar enough, the classifier trained on the labeled set can achieve similar performance compared to the classifier trained on the entire training dataset with all samples annotated. Therefore, besides the uncertainty, dd-AL also evaluates each unlabeled sample's effectiveness in decreasing the distribution discrepancy between the labeled set and the unlabeled set after we annotate it, which is further represented by the representativeness and rarity evaluation metrics.

2.2. METHOD

The workflow of our deep active learning framework is illustrated in Figure 2.1. In each annotation suggestion stage, we first pass each unlabeled sample through K sag-FCNs to obtain its averaged feature representation and K segmentation probability maps. Then, based on the feature representation and segmentation probability maps of each unlabeled sample, dd-AL selects most valuable unlabeled samples based on their uncertainty to the sag-FCNs and effectiveness in decreasing the data distribution discrepancy between the labeled and unlabeled set. Finally, these selected samples will be annotated and sent to the sag-FCN for the supervised training. We conduct this annotation suggestion process iteratively until satisfied.

2.2.1. Semantic Attention Guided Fully Convolutional Network. Based on recent advances of deep neural network structures such as residual networks Wang *et al.* (2017a) and non-local networks Wang *et al.* (2018), we propose a semantic attention guided fully convolutional network that automatically highlights the feature activation related to the target object using our proposed semantic attention guidance units. In addition, we carefully design the architecture of the sag-FCN to reduce its parameter space while maintaining the good generalization capability, suitable for the active learning.

Compared with the original Fully Convolutional Network (FCN, Long *et al.* (2015)), the proposed attention guided Fully Convolutional Network (sag-FCN), shown in Figure 2.2, has three significant improvements:

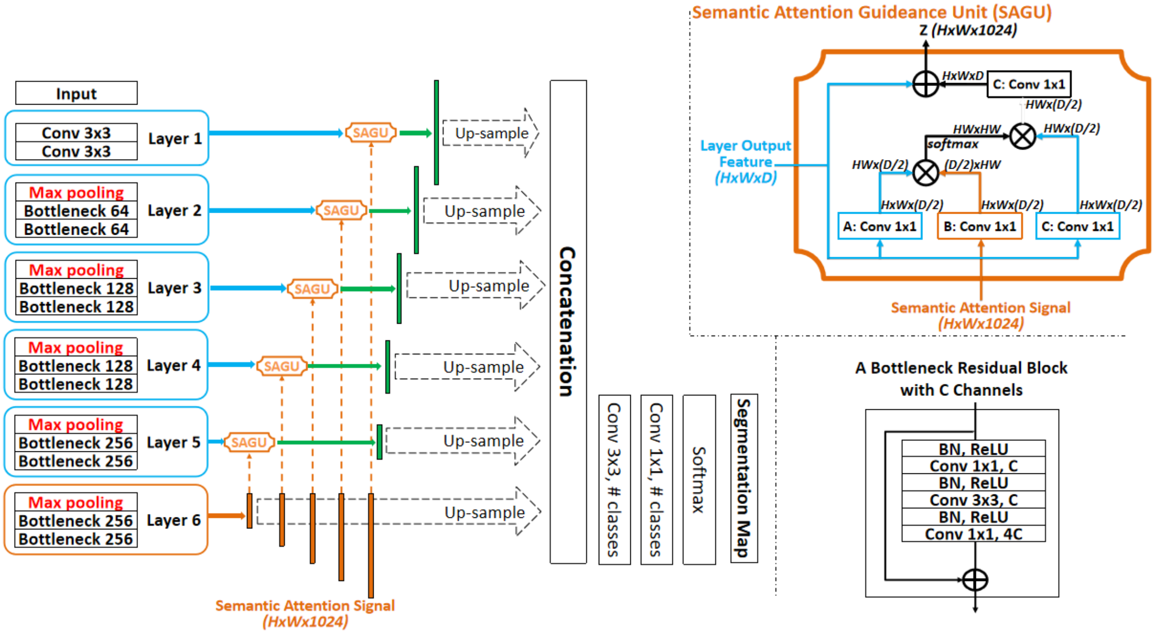


Figure 2.2. The architecture of the proposed semantic attention guided fully convolutional network.

Semantic attention guidance unit: We propose the Semantic Attention Guidance Unit (SAGU) to fuse the high-level semantic features to low- and mid-level features. SAGU exploits the high-level semantic information as soft self-attentions that lead low- and mid-level features to focus on target areas and highlight the feature activations that are relevant to the target instance. Therefore, SAGUs ensure that the sag-FCN can conduct accurate segmentation on object instances with high variabilities.

Feature fusion strategy: Compared with the conventional skip-connections that progressively merge low-level features to the up-sampling process of high-level features Long *et al.* (2015), the feature fusion strategy in the sag-FCN considers each layer's attentive features (with semantic attention) as an up-sampling "seed". All "seeds" will be progressively up-sampled to the input image size, and then be concatenated for generating smooth segmentation results.

Bottleneck residual modules: In sag-FCN, we replace most convolutional layers by bottleneck residual modules to significantly reduce the number of parameters while maintaining the same receptive field size and feature channels at the end of each module. This design reduces the training cost with less parameters (i.e., suitable for iterative active learning) and maintains sag-FCN's generalization capability.

These three improvements of our sag-FCN are essential when combining deep neural networks and active learning. First, the performance of sag-FCN using all training data is the upper bound of the performance of our deep active learning framework that uses only a portion of the training data. By using our SAGUs and feature fusion strategy, the proposed sag-FCN can achieve state-of-the-art segmentation performance using all training data, which provides a good upper bound for our framework. Second, at the start of active learning, since only a few training samples are available, the model that has too many free parameters will be hard to train. Hence, we use the bottleneck residual blocks to significant reduce the numnber of parameter without decreasing the number of feature channels, which allows the sag-FCN to have good generalization capability to produce reasonable results when very little training data is available.

2.2.2. Distribution Discrepancy Based Active Learning Algorithm. In general, our distribution discrepancy based active learning algorithm (dd-AL) suggests samples for annotation based on two criteria: (1) the uncertainty to the segmentation network and (2) the effectiveness in decreasing the distribution discrepancy between the labeled set and unlabeled set. Since parallelly evaluating these two criteria of each unlabeled sample is computational expensive, our dd-AL conducts the annotation suggestion process in two sequential steps. As shown in Figure 2.1, in the first step, dd-AL selects N^c samples with the lowest uncertainty scores from the unlabeled set as candidate samples. In the second step, among these N^c candidate samples, dd-AL selects a subset of them that has the highest effectiveness in decreasing the distribution discrepancy between the labeled set and unlabeled set.

2.2.2.1. Evaluating a sample's uncertainty. In the first step of dd-AL, to evaluate the uncertainty of each unlabeled sample, we adopt the bootstrapping strategy that trains K sag-FCNs, each of which only uses a subset of the suggested data for training in each annotation suggestion stage, and calculates the disagreement among these K models. Specifically, in each annotation suggestion stage, for each unlabeled sample s^u whose spatial dimension is $h \times w$, we first use K sag-FCNs to generate K segmentation probability maps of s^u. Then, we compute an uncertainty score $u_k^{s^u}$ of the k-th ($k \in [1, K]$) segmentation probability map of s^u by using the Best-versus-Second-Best (BvSB) strategy Joshi *et al.* (2012):

$$u_k^{s^u} = \frac{1}{h \times w} \sum_{i=1}^{h \times w} (1 - \left| p_{k,i}^{best} - p_{k,i}^{second} \right|), \tag{2.1}$$

where $p_{k,i}^{best}$ and $p_{k,i}^{second}$ denote the probability values of the best class guess and the second best class guess of the i-th pixel on s^u, respectively, predicted by the k-th sag-FCN. $(1 - \left| p_{k,i}^{best} - p_{k,i}^{second} \right|)$ denotes the pixel-wise BvSB score, where a larger score indicates more uncertainty. In Eq. 2.1, the uncertainty score of an image sample s^u is the average of the BvSB scores of all pixels in this image.

Finally, we compute the uncertainty score of s^u by averaging the uncertainty scores predicted by the K sag-FCNs:

$$u_{final}^{s^u} = \frac{1}{K} \sum_{k=1}^{K} u_k^{s^u}. \tag{2.2}$$

We rank all the unlabeled image samples based on their uncertainty scores and select the top N^c samples with the highest uncertainty scores as the candidate set S^c for the second step of dd-AL.

2.2.2.2. Evaluating a sample's effectiveness in decreasing discrepancy. In the second step of dd-AL, we intend to annotate a few candidate samples in the candidate set S^c that can help us achieve the smallest distribution discrepancy between the labeled set and unlabeled set after the annotation, which also means annotating these candidate samples can make the distributions of these two sets more similar compared to annotating the other candidate samples. After several annotation suggestion stages, if the distributions of the labeled set and unlabeled set are similar enough, the classifier trained on the labeled set can achieve similar performance compared to the classifier trained on the entire dataset with all samples annotated.

In each annotation suggestion stage, we define S^l as the labeled set with N^l samples and S^u as the unlabeled set with N^u samples. We use the i-th candidate sample s_i^c in S^c, where $i \in [1, N^c]$, as a reference data point to estimate the data distributions of the unlabeled set S^u and the labeled set S^l, and compute a distribution discrepancy score d_i^c that represents the distribution discrepancy between S^u and S^l after annotating s_i^c:

$$d_i^c = \frac{1}{N^l + 1} \sum_{j=1}^{N^l+1} Sim(s_i^c, s_j^l) - \frac{1}{N^u - 1} \sum_{j=1}^{N^u-1} Sim(s_i^c, s_j^u). \tag{2.3}$$

In Eq. 2.3, the first term represents the data distribution of the labeled set S^l estimated by s_i^c, where $Sim(s_i^c, s_j^l)$ represents the *cosine similarity* between s_i^c and the j-th sample s_j^l in the labeled set S^l in the high-dimensional feature space[1]. The second term in Eq. 2.3 represents the data distribution of the unlabeled set S^u estimated by s_i^c, where $Sim(s_i^c, s_j^u)$ represents the *cosine similarity* between s_i^c and the j-th sample s_j^u of the unlabeled set S^u in the high-dimensional feature space. After we compute the distribution discrepancy scores for all candidate samples in S^c, the candidate sample with the lowest score can be consider as the most valuable sample for the annotation.

To accelerate the annotation suggestion process, we prefer to suggest multiple samples for the annotation in each stage instead of suggesting one sample at a time. However, directly ranking the candidate samples in a descending order based on their distribution

[1] The encoding part of each sag-FCN can be utilized as a feature extractor. Given an input image to K sag-FCNs, the average of outputs of Layer 6 in these sag-FCNs can be viewed as a high-dimensional feature representation of the input image.

discrepancy scores and suggesting the top ones is inaccurate. Since the distribution discrepancy of the labeled and unlabeled sets is computed based on annotating one sample at a time.

To address this problem, we propose the idea of super-sample s^{super}, which is a m-combination of the candidate set S^c with N^c samples. In total, there are $\binom{N^c}{m}$ possible super-samples that can be generated from S^c. The feature representation of each super-sample is the average of the feature representations of the m samples within it. Thus, we can rewrite the distribution discrepancy score computation in Eq. 2.3 into a super-sample version as:

$$d_q^{super} = \frac{1}{N^l + m} \sum_{j=1}^{N^l+m} Sim(s_q^{super}, s_j^l) - \frac{1}{N^u - m} \sum_{j=1}^{N^u-m} Sim(s_q^{super}, s_j^u), \tag{2.4}$$

where d_q^{super} denotes the distribution discrepancy score of the q-th super-sample s_q^{super} in the candidate set S^c. Then, the super-sample with the lowest distribution discrepancy score will be suggested, where the m samples within this super-sample will be the final suggested samples in this annotation suggestion stage. These samples with their annotations will be used to train the sag-FCNs.

The suggestion is to find super-sample with the lowest distribution discrepancy score in Eq. 2.4. In other words, dd-AL intends to suggest samples that can minimize the first term in Eq. 2.4, which is equivalent to minimizing the similarity between suggested samples and the labeled set S^l. Therefore, the proposed dd-AL ensures the high rarity of suggested samples in the labeled set. Also, in Eq. 2.4, dd-AL intends to suggest samples that can maximize the second term, which is equivalent to maximizing the similarity between suggested samples and the unlabeled set S^l. Therefore, the proposed dd-AL can also ensure the high representativeness of suggested samples in the unlabeled set.

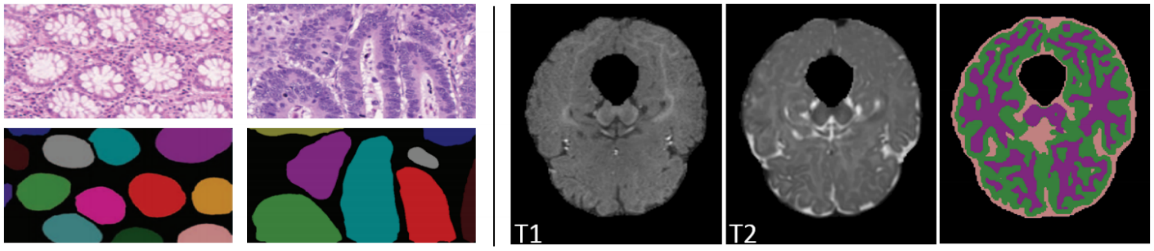


Figure 2.3. Some qualitative results of our framework on GlaS dataset (left) and iSeg dataset (right, pink: Cerebrospinal Fluid; purple: White Matter; green: Gray Matter) using only 50% training data.

2.3. EXPERIMENT

In this section, we validate the effectiveness of our ag-FCN and our entire deep active learning framework (ag-FCN + dd-AL) on two challenging datasets.

2.3.1. Dataset. We use the 2015 MICCAI gland segmentation dataset (GlaS, Sirinukunwattana *et al.* (2017)) and the training set of 2017 MICCAI infant brain segmentation dataset (iSeg, Wang *et al.* (2019)) to evaluate the effectiveness our deep active learning framework. The GlaS dataset contains 85 training images and 80 testing images (Test A: 60 images; Test B: 20 images.). The training set of iSeg dataset contains T1- and T2-weighted MR images of 10 infant subjects. We enlarge the training data using data augmentation techniques, including rotation, flipping, elastic distortion and random cropping.

2.3.2. Implementation Details. We train 3 sag-FCNs ($K = 3$) for 1600 stages and 1000 stages for the GlaS dataset and iSeg dataset, respectively. For each stage, we select top 10 uncertain samples in the first step of dd-AL ($N^c = 10$), and finally suggest one super-sample that contains 6 samples ($m = 6$) for annotation in the second step of dd-AL. At the end of each stage, sag-FCNs will be trained with all available labeled data.

2.3.3. Experiments on GlaS Dataset. We first compared our sag-FCNs using all training data with state-of-the-art methods. As shown in Table 2.1, our sag-FCNs achieves very competitive segmentation performances (best in five columns, second best in one column), which shows the effectiveness of our sag-FCN and attention gate unit in producing

Table 2.1. Comparison with state-of-the-art methods on 2015 MICCAI Gland Segmentation challenge dataset.

Method	F1 Score		ObjectDice		ObjectHausdorf	
	Test A	Test B	Test A	Test B	Test A	Test B
CUMedNet Chen *et al.* (2016)	0.912	0.716	0.897	0.781	45.418	160.347
MILD-Net Graham *et al.* (2019)	0.920	0.820	0.918	0.836	**39.390**	103.070
FCN-MCS Yang *et al.* (2017)	0.921	0.855	0.904	0.858	44.736	96.976
Ours (full training data)	**0.937**	**0.866**	**0.926**	**0.871**	40.812	**95.328**
FCN-MCS Yang *et al.* (2017) (50% training data)	0.913	0.832	0.901	0.836	-	-
Ours (50% training data)	**0.924**	**0.851**	**0.912**	**0.853**	43.728	101.873

accurate pixel-wise predictions on biomedical images. To validate our deep active learning framework (sag-FCNs and dd-AL), we simulate the annotation suggestion process by only providing the suggested samples and their annotations to the sag-FCNs for training. We consider the annotation cost as the number of annotated pixels and set the annotation cost budget as 10%, 30% and 50% of the overall labeled pixels. Our framework is compared with (1) Random Query: randomly selecting image samples until reaching the budget; (2) Uncertainty Query: suggesting samples only considering the uncertainty evaluation (the first step of dd-AL); (3) MCS, a minimum cover set based active learning algorithm that only considers the uncertainty and representativeness information proposed in Yang *et al.* (2017). As shown in Figure 2.4, our framework, which not only considers the suggested sample's uncertainty, representativeness and rarity but also progressively decreases the distribution discrepancy between the unlabeled set and labeled set, is consistently better than the other three methods. As shown in Table 2.1, our framework can achieve state-of-the-art performance using only 50% of the training data.

2.3.4. Experiments on iSeg Dataset. We extend the proposed sag-FCN into the 3D version (3D-sag-FCN)[2] and test our deep active learning framework (3D-sag-FCN and dd-AL) on the training set of iSeg dataset using 10-fold cross-validation (9 subjects for training, 1 subject for testing, repeat 10 times). As shown in Table 2.2, our framework can achieve competitive performances only using 50% training data.

[2] To extend the sag-FCN shown in Fig. 2.2 into a 3D version, we replace all 2D operations with 3D operations (e.g., replacing 2D convolutions with 3D convolutions, etc.).

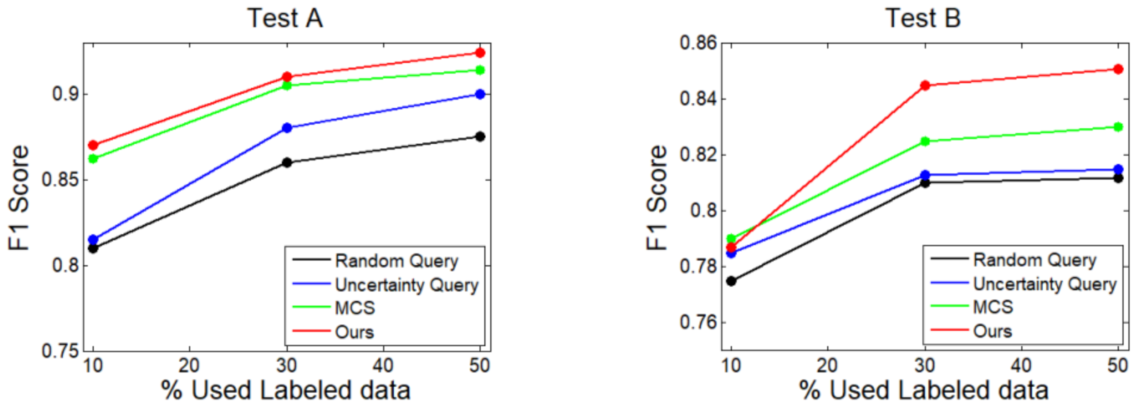


Figure 2.4. Comparison using limited training data of GlaS dataset.

2.4. SUMMARY

To significantly alleviate the burden of manual labeling in the biomedical image segmentation task, in this work, we propose a deep active learning framework that consists of: (1) a semantic attention guided fully convolutional network (sag-FCN) that achieves state-of-the-art segmentation performances when using the full training data and (2) a distribution discrepancy based active learning algorithm that progressively suggests valuable samples to train the sag-FCNs. Our proposed framework can achieve state-of-the-art segmentation performance by only using 50% of the annotated training data.

2.5. RETHINKING

Similar to the existing attention mechanisms, the proposed Semantic Attention Guidance Unit (SAGU) is also a high-level feature dominant algorithm, where the semantic guidance process in SAGU is passive and indiscriminate in the low- or mid-level feature's perspective. Although the SAGU can effectively improve the performance of deep neural networks in biomedical image segmentation tasks, we still have an unanswered research question: *do different low- or mid-level features need the same semantic attention as their guidance?*

Table 2.2. Comparison with state-of-the-art methods on iSeg training set.

Method	DICE		
	White Matter	Gray Matter	Cerebrospinal Fluid
3D-Unet Çiçek *et al.* (2017)	0.896	0.907	0.944
3D-DenseNet Bui *et al.* (2017)	0.913	0.916	0.947
Ours (full training data)	**0.927**	**0.921**	**0.959**
Ours (50% training data)	0.909	0.912	0.951

In the biomedical image analysis, in which the high-level semantics of the image is always straightforward and clear, it is intuitive and effective to capture the high-level semantics and indiscriminately distribute them to all low- or mid-level features for semantic attention guidance. However, in natural images, the high-level semantics are always sophisticated. The semantic attention guidance for natural images might need to consider more flexible and discriminative correlations between high-level semantics and low/mid-level features. Moreover, in the natural image sequence, the high-level semantics of each frame is not only sophisticated but also highly correlated to other frames. Due to this reason, capturing high-level semantics over the entire image sequence remains a challenging task to existing attention mechanisms.

Motivated by the above research question and its challenging extension in natural image sequences, we try to propose an attention framework to solve it in a challenging computer vision task, the video-based person re-identification, in Section 2.

3. ZOOM IN AND OUT: TWO-LEVEL SPATIAL AND TEMPORAL ATTENTION NETWORK FOR VIDEO-BASED PERSON RE-IDENTIFICATION

3.1. RESEARCH BACKGROUND

Person re-identification (re-id), which is firstly researched in the image domain to match images of the same individual across multiple non-overlapping cameras, remains a challenging task due to the dramatic variations of the human appearance and pose, background distraction, and occlusion. Also, in real world scenarios, the candidates to be matched are typically collected by pedestrian detectors that might generate imprecise person bounding boxes, leading to the misalignment challenge in the re-id task Gong *et al.* (2013).

As an improvement of the image-based re-id task, the video-based person re-id task, which matches video sequences of the same individual, provides more information relevant to a person's appearance, gait, and motion over time. Benefited from the advances of Convolutional Neural Networks (CNNs), recent video-based re-id works tend to use CNNs to obtain high-level features from each frame of the video sequence and then merge them by concatenation or feature embedding Liu *et al.* (2017); Zeng *et al.* (2018). Meanwhile, some work tries to apply recurrent neural networks with the extracted CNN features to capture the long-range temporal dependencies McLaughlin *et al.* (2016b); Xu *et al.* (2017).

3.1.1. Challenges and Motivations. Although video-based re-id has more visual characteristics to describe a person, unfortunately it not only inherits all aforementioned challenges in the image domain but also extends them to the temporal domain as the *temporal information inconsistency challenge* (i.e., the dramatic variations of human appearance and shape, the misalignment of person detection boxes, and the occlusion, causing a person's

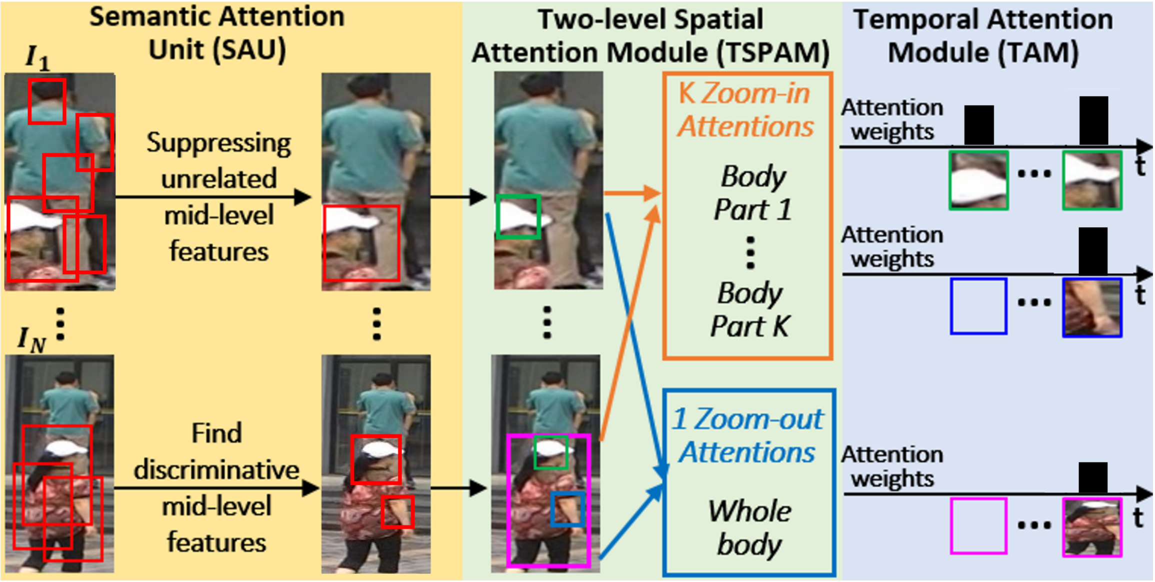

Semantic Attention Unit (SAU)
I_1
Suppressing unrelated mid-level features
I_N
Find discriminative mid-level features
Two-level Spatial Attention Module (TSPAM)
K Zoom-in Attentions
Body Part 1
Body Part K
1 Zoom-out Attentions
Whole body
Temporal Attention Module (TAM)
Attention weights
t
Attention weights
t
Attention weights
t

Figure 3.1. To overcome the temporal information inconsistency challenge, we propose a novel framework that consists of three main components: SAU, TSPAM and TAM.

body parts inconsistently visible in the video sequence). Despite the recent advance in video-based re-id, this temporal inconsistency challenge is still not well solved in two aspects:

(1) Most of the state-of-the-art methods rely on the high-level features extracted from the last convolutional layer of CNN. Given an input image, CNN first collects fine local details (low-level features), then progressively associates these details into a general understanding (high-level features) of the entire image. During this process, a lot of regional information in the mid-level is pooled to allow CNN to have a good generalization capability of accomplishing vision tasks based on the image's semantic information. However, the global semantic information of the whole human body captured by high-level features is not sufficient to perform the person re-id task in which the regional body parts may be discriminative to identify persons, especially during occlusions. In the meantime, the mid-level regional feature is blind to the global semantic information, and there is no clue on which mid-level features are most relevant to human body. Therefore, the motivated

research question is: *from the feature maps extracted from CNN, how can we exploit both the high-level semantic information and the mid-level regional information related to human body parts for the video-based person re-id task?*

(2) In real world videos, the target person may be frequently occluded from the camera, thus his/her whole body may be visible only in a portion of the video and some frames may contain only a few body parts. Moreover, due to the misaligned bounding boxes from person detectors, some frames may contain a small portion of the target person but a large portion of another person (e.g., Figure 3.1). Videos that contain these "problematic" frames will result in inferior re-id performances. Existing methods tend to address this issue by assigning a weight to each frame to represent the feature importance of the frame in the video. For instance, those "problematic" frames will be considered as less or not important and assigned very low scores (or zero scores). However, within these "problematic" frames, the remaining visible body parts of the target person may contain strong cues from different viewpoints for the re-id task, which are ignored by these methods. Therefore, the motivated research question is: *given a video that contains frames with occlusions and misalignments, how can we extract image features with the attention paid to the visible whole body or body parts for the video-based re-id task?*

3.1.2. Our Proposal and Contribution. Motivated by these two research questions, we propose a novel end-to-end video-based person re-id framework (Figure 3.1) which consists of the following key components:

- A Semantic Attention Unit (SAU) that extracts mid-level features from CNN to describe body parts. By selectively bringing high-level semantics to the mid-level features, SAU suppresses the mid-level features unrelated to the re-id task.
- A Two-level Spatial Attention Module (TSPAM) that applies K *zoom-in attentions* on the mid-level features of each frame to capture discriminative body parts, and one *zoom-out attention* on the high-level feature of each frame to capture the whole visible

body. Within the TSPAM, we introduce a constrained diversity regularization, which ensures multiple *zoom-in attentions* do not repeatedly discover the same body part and all K body parts are associated with the target person.

- A Temporal Attention Module (TAM) that assigns $K + 1$ temporal attention weights to each frame to represent the feature importances of the K body parts and the whole body. All feature vectors from individual frames are aggregated into an overall feature representation that contains all spatiotempral information of the target person in the input video for the re-id task.

3.2. RELATED WORK

As an extension of image-based person re-identification (re-id), video-based re-id is intuitively closer to the practical scenario as video streams are continuously captured by surveillance cameras Xu *et al.* (2017). However, due to the temporal information inconsistency challenge, video-based re-id is still a difficult unsolved task.

3.2.1. Deep Learning for Video-based Re-id. Video-based re-id recently has achieved notable progresses by using deep learning models. For example, Liu *et al.* (2017) and Zeng *et al.* (2018) design deep neural networks for fusing frame-wise features into one overall feature representation of the person in a video. To better capture the long-range temporal dependencies, McLaughlin *et al.* (2016a) and Yan *et al.* (2016) exploit recurrent convolutional networks to encode significant temporal features of the video and compute the feature similarity of video pairs. These methods utilize deep neural networks as high-level feature extractors without applying attention mechanisms.

3.2.2. Attention Models for Video-based Re-id. Attention mechanism, which learns attention masks to highlight important features, has been widely used in re-id tasks in recent years Fu *et al.* (2019); Li *et al.* (2018b). Song *et al.* (2018) apply a separately trained image segmentation model to provide a mask of the human body, enforcing the

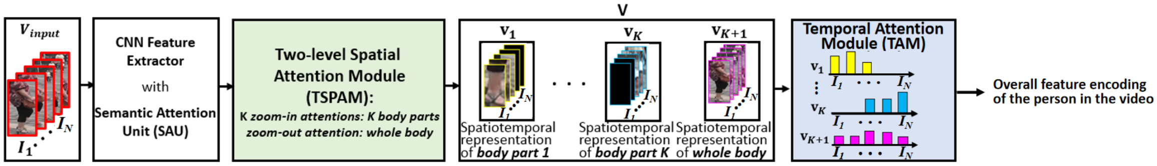

V_{input}
I_1
I_N
CNN Feature Extractor
with
Semantic Attention Unit (SAU)
Two-level Spatial Attention Module (TSPAM):
K zoom-in attentions: K body parts
zoom-out attention: whole body
V
v_1
v_K
v_{K+1}
Spatiotemporal representation of body part 1
Spatiotemporal representation of body part K
Spatiotemporal representation of whole body
Temporal Attention Module (TAM)
Overall feature encoding of the person in the video

Figure 3.2. The workflow of our video-based person re-id framework.

spatial attention to focus on significant body parts within the mask. Li *et al.* (2018a) propose a deep attention architecture to highlight different body parts over time. Liu *et al.* (2019) propose a stacked non-local network to capture characteristic spatial and temporal features of the target person's whole body. Hu *et al.* (2018) introduce squeeze and excitation networks, putting attentions on different feature channels. Most of these attention models generate attention maps from the high-level features extracted from CNN. Different from these methods, our network employs spatial and temporal attention on top of both the mid- and high-level feature maps to learn a latent feature vector that represents the comprehensive information of the person and emphasizes discriminative regional information of the body parts.

3.3. METHOD

The workflow of our framework is illustrated in Figure 3.2. For each frame of an input video, we first design a Semantic Attention Unit (SAU, Figure 3.3) to extract mid-level features from the intermediate convolutional layer of CNN, and high-level features from the last convolutional layer of CNN. Then, we design a Two-level Spatial Attention Module (TSPAM, Figure 3.4) that applies K *zoom-in attentions* to extract image feature vectors that represent K discriminative body parts from the mid-level feature map, and applies one *zoom-out attention* to extract one image feature vector that represents the whole body from the high-level feature map. Meanwhile, we introduce a constrained diversity regularization to encourage the *zoom-in attentions* to not only focus on different body parts

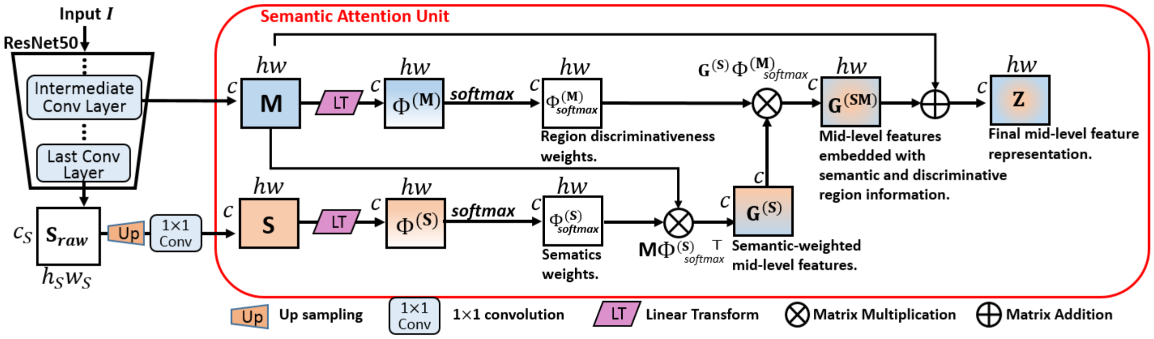


Figure 3.3. The computational graph of the Semantic Attention Unit.

but also be highly related to the *zoom-out attention*. Thirdly, for each body part (and the whole body), we design a Temporal Attention Module (TAM, Figure 3.5) to pool its image feature vectors from individual frames across the duration of the video to generate a feature representation that represents the body parts and whole body in the entire video. The learned feature vectors of all body parts and the whole body will be aggregated and sent to a fully connected layer that represents the final encoding of the target person in the input video. A batch-hard triplet loss and a cross entropy loss are combined with the constrained diversity regularization to train the whole network in an end-to-end fashion.

3.3.1. Semantic Attention Unit. Instead of blindly pooling mid-level feature maps to obtain the regional information, our method employs a Semantic Attention Unit (SAU, Figure 3.3) to extract mid-level features that are discriminative and related to the target person.

We adopt the ResNet50 CNN architecture Wang *et al.* (2017b) for extracting image features from each frame of the video. The CNN starts with a convolutional layer (*conv1*), followed by four residual blocks (*res2*, *res3*, *res4* and *res5*). We exploit the output of *res2*, *res3* or *res4* as mid-level features[1] and the output of *res5* as high-level features.

[1] The sensitivity study on which residual block's output is extracted as the mid-level feature is in the *Experiment* section.

Given a frame I of the video V to the CNN, let $\mathbf{M} \in \mathbb{R}^{c \times hw}$ and $\mathbf{S}_{raw} \in \mathbb{R}^{c_S \times h_S w_S}$ denote the mid- and high-level feature maps of I, respectively, where c and c_S denote the number of feature channels, and hw and $h_S w_S$ are the vectorized spatial dimensions. We first apply the up-sampling operation followed by a 1×1 convolution on $\mathbf{S}_{raw}$ to obtain $\mathbf{S} \in \mathbb{R}^{c \times hw}$ that has the same dimension as $\mathbf{M}$.

Then, SAU uses the high-level feature $\mathbf{S}$ to guide the extraction of mid-level features with semantic meanings. Two attention weight matrices $\mathbf{W}^{(S)} \in \mathbb{R}^{c \times hw}$ and $\mathbf{B}^{(S)} \in \mathbb{R}^{c \times hw}$ are learned to compute a semantic attention score matrix $\mathbf{\Phi}^{(S)}$, which indicates the feature importance of different image regions in $\mathbf{S}$ to represent the semantic meaning:

$$\mathbf{\Phi}^{(S)} = \mathbf{W}^{(S)} \circ \mathbf{S} + \mathbf{B}^{(S)}, \quad \text{where} \quad \mathbf{\Phi}^{(S)} \in \mathbb{R}^{c \times hw}, \tag{3.1}$$

where $\circ$ is the element-wise product. The semantic attention score matrix $\mathbf{\Phi}^{(S)}$ is normalized by applying *softmax* operation in every row, generating $\mathbf{\Phi}^{(S)}_{softmax}$.

Now, we apply the high-level semantic attention weights $\mathbf{\Phi}^{(S)}_{softmax}$ to the mid-level feature maps $\mathbf{M}$:

$$\mathbf{G}^{(S)} = \mathbf{M}\mathbf{\Phi}^{(S)}_{softmax}{}^{\top}, \quad \text{where} \quad \mathbf{G}^{S} \in \mathbb{R}^{c \times c}. \tag{3.2}$$

$\mathbf{G}^{(S)}$ can be considered as a mid-level feature collection weighted by the semantic information from $\mathbf{S}$.

Instead of indiscriminately distributing the entire collection $\mathbf{G}^{(S)}$ to all regions in the mid-level feature map $\mathbf{M}$, we further propose to investigate the discriminativeness of each region in $\mathbf{M}$. Specifically, we first learn attention weight matrices $\mathbf{W}^{(M)} \in \mathbb{R}^{c \times hw}$ and $\mathbf{B}^{(M)} \in \mathbb{R}^{c \times hw}$, and compute a spatial attention score matrix $\mathbf{\Phi}^{(M)}$ that indicates the feature importance for representing regional information from different spatial regions in $\mathbf{M}$:

$$\mathbf{\Phi}^{(M)} = \mathbf{W}^{(M)} \circ \mathbf{M} + \mathbf{B}^{(M)}, \quad \text{where} \quad \mathbf{\Phi}^{(M)} \in \mathbb{R}^{c \times hw}. \tag{3.3}$$

The spatial attention score matrix $\mathbf{\Phi}^{(M)}$ is passed through a *softmax* layer to get the region discriminativeness weights $\mathbf{\Phi}^{(M)}_{softmax} \in [0, 1]^{c \times hw}$. Next, we apply $\mathbf{\Phi}^{(M)}_{softmax}$ to the semantic-weighted mid-level feature $\mathbf{G}^{(S)}$:

$$\mathbf{G}^{(SM)} = \mathbf{G}^{(S)} \mathbf{\Phi}^{(M)}_{softmax}, \tag{3.4}$$

where $\mathbf{G}^{SM} \in \mathbb{R}^{c \times hw}$ is the mid-level feature representation that contains not only the global semantic information of the target object from $\mathbf{S}$ but also the discriminative regional information from $\mathbf{M}$.

Finally, in addition to the above multiplicative attention, we can have one more step of additive attention as:

$$\mathbf{Z} = \mathbf{G}^{(SM)} + \mathbf{M} \tag{3.5}$$

where $\mathbf{Z} \in \mathbb{R}^{c \times hw}$ is the output of SAU as the final mid-level feature representation of the frame I.

Note, in SAU, the implementation of Eq. 3.2 can be further explained by the Semantic Attention Guidance Unit (SAGU) proposed in the sag-FCN in Section 2 for guiding mid-level features with high-level semantic attentions. However, after obtaining the semantically guided feature collection $\mathbf{G}^{(S)}$ in Eq. 3.2, the SAU takes SAGU a step further by introducing Eq. 3.4 that selectively distributes $\mathbf{G}^{(S)}$ to $\mathbf{M}$. This mechanism allows SAU to capture more flexible and discriminative correlations between mid- and high-level features than SAGU that *indiscriminately* distributes high-level semantics to *all* regions in $\mathbf{M}$[2].

3.3.2. Two-level Spatial Attention Module. We design a Two-level Spatial Attention Module (TSPAM, Figure 3.4) to discover different salient body parts (*zoom-in attentions*) from the mid-level feature and highlight the whole body (*zoom-out attention*) from the high-level feature.

[2] We carry out experiments to compare the SAU and SAGU in the *More Analysis on Diversity Regularization and Semantic Attention Unit* section of the *Experiment* section.

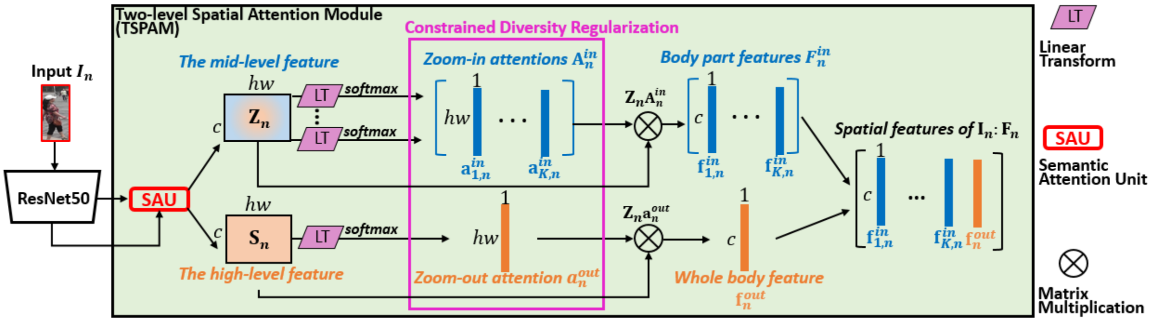


Figure 3.4. The computational graph of the Two-level Spatial Attention Module on an arbitrary frame n.

Given a video V that contains N frames, by using a pre-trained CNN and our proposed SAU, we encode each frame I_n into a mid-level feature representation $\mathbf{Z}_n \in \mathbb{R}^{c \times hw}$ and a high-level feature representation $\mathbf{S}_n \in \mathbb{R}^{c \times hw}$, where $n \in [1, N]$ denotes the frame number in video V. In the following we design *zoom-in attentions* and *zoom-out attention* to extract image features of discriminative body parts and the whole human body from $\mathbf{Z}_n$ and $\mathbf{S}_n$, respectively.

3.3.3. Zoom-in Attentions. To compute the *zoom-in attentions* of I_n, we use K spatial attention models to capture the significant body parts of the target person from the mid-level feature $\mathbf{Z}_n$. For the k-th attention model, where $k \in [1, K]$, we first learn attention weight vectors $\mathbf{w}_{k,n}^{(Z_n)} \in \mathbb{R}^{c \times 1}$ and $\mathbf{b}_{k,n}^{(Z_n)} \in \mathbb{R}^{hw \times 1}$, and compute a spatial attention score vector $\boldsymbol{\phi}^{(Z_n)}$, which indicates the feature importance for representing a body part from different spatial regions on $\mathbf{Z_n}$:

$$\boldsymbol{\phi}^{(Z_n)} = (\mathbf{Z}_n)^\top \mathbf{w}_{k,n}^{(Z_n)} + \mathbf{b}_{k,n}^{(Z_n)}, \quad \text{where} \quad \boldsymbol{\phi}^{(Z_n)} \in \mathbb{R}^{hw \times 1}. \tag{3.6}$$

Then, we pass the spatial attention score vector $\boldsymbol{\phi}^{(Z_n)}$ through a *softmax* layer to get the k-th *zoom-in attention*, $\mathbf{a}_{k,n}^{in} \in [0, 1]^{hw \times 1}$, which localizes a body part of the target person. Thus, for each frame I_n in the video V, we obtain K *zoom-in attentions* ensembled in a matrix $\mathbf{A}_n^{in} = [\mathbf{a}_{1,n}^{in}, \ldots, \mathbf{a}_{K,n}^{in}] \in \mathbb{R}^{hw \times K}$. Finally, all the body part features in frame n, $\mathbf{Z}_n$, are weighted by their corresponding attentions:

$$\mathbf{F}_n^{in} = \mathbf{Z}_n \mathbf{A}_n^{in}, \tag{3.7}$$

where $\mathbf{F}_n^{in} = [\mathbf{f}_{1,n}^{in}, ..., \mathbf{f}_{K,n}^{in}] \in \mathbb{R}^{c \times K}$ and $\mathbf{f}_{k,n}^{in}$ represents the image feature vector of the body part discovered by the k-th *zoom-in attention* $\mathbf{a}_{k,n}^{in}$.

3.3.4. Zoom-out Attention. To compute the *zoom-out attention*, we use one spatial attention model to capture the entire visible human body from the high-level feature $\mathbf{S}_n$. We learn attention weight vectors $\mathbf{w}_n^{(S_n)} \in \mathbb{R}^{c \times 1}$ and $\mathbf{b}_n^{(S_n)} \in \mathbb{R}^{hw \times 1}$, and compute a spatial attention score vector $\boldsymbol{\phi}^{(S_n)}$, which indicates the feature importance for representing the whole body from different spatial regions on $\mathbf{S}_n$:

$$\boldsymbol{\phi}^{(S_n)} = (\mathbf{S}_n)^\top \mathbf{w}_n^{(S_n)} + \mathbf{b}_n^{(S_n)}, \text{where } \boldsymbol{\phi}^{(S_n)} \in \mathbb{R}^{hw \times 1}. \tag{3.8}$$

Then, we pass the spatial attention score vector $\boldsymbol{\phi}^{(S_n)}$ through a *softmax* layer to get the *zoom-out attention*, $\mathbf{a}_n^{out} \in [0, 1]^{hw \times 1}$, which localizes the whole body of the target person. Finally, we denote $\mathbf{f}_n^{out}$ as a feature vector that represents the whole body captured by *zoom-out attention* $\mathbf{a}_n^{out}$ in frame I_n. The $\mathbf{f}_n^{out}$ is computed by

$$\mathbf{f}_n^{out} = \mathbf{Z}_n \mathbf{a}_n^{out}. \tag{3.9}$$

By applying the TSPAM on frame I_n, all significant body parts discovered by *zoom-in attentions* and the whole body with visible regions discovered by *zoom-out attention* are concatenated as a feature representation:

$$\mathbf{F}_n = [\mathbf{F}_n^{in}, \mathbf{f}_n^{out}] = [\mathbf{f}_{1,n}^{in}, ..., \mathbf{f}_{K,n}^{in}, \mathbf{f}_n^{out}], \qquad \mathbf{F}_n \in \mathbb{R}^{c \times (K+1)}. \tag{3.10}$$

3.3.5. Constrained Diversity Regularization. Without any constraints, *zoom-in attentions* may easily discover the same significant body part in the same frame. It is natural to apply diversity regularization terms to diversify these *zoom-in attentions* to different body parts, so it is robust to occlusion Li *et al.* (2018a); Lin *et al.* (2017). However, we also need to ensure these body parts are highly related to the target person other than some other persons who either occlude the target person or appear at the background. Therefore, we design a penalty term, the *c*onstrained *diver*sity regularization (Reg_{cdiver}), to ensure the uniqueness of each *zoom-in attention* as well as the correlation between *zoom-in attentions* and the *zoom-out attention* within the frame I_n:

$$Reg_{cdiver} = \sum_{k=1}^{K} \frac{1}{\sqrt{2}} \left\| \sqrt{\mathbf{a}_{k,n}^{in}} - \sqrt{\mathbf{a}_{n}^{out}} \right\|_2 - \sum_{k_1=1}^{K} \sum_{k_2=k_1+1}^{K} \frac{1}{\sqrt{2}} \left\| \sqrt{\mathbf{a}_{k_1,n}^{in}} - \sqrt{\mathbf{a}_{k_2,n}^{in}} \right\|_2 . \tag{3.11}$$

In Eq. 3.11, the first term, which computes the sum of the Hellinger distance Beran (1977) between each *zoom-in attention* and the *zoom-out attention*, intends to be minimized to ensure that all *zoom-in attentions* (body parts) are associated with the *zoom-out attention* (the target person). The second term in Eq. 3.11, which computes the sum of the Hellinger distance between every two *zoom-in attentions* in A_n^{in}, intends to be maximized to ensure the dissimilarity between every two *zoom-in attentions* is large.

3.3.6. Temporal Attention Module. After extracting image feature vectors that either represent individual body parts or the whole body from individual frames of the video, we need to consider how to combine these feature vectors to generate an overall feature representation for the target person over the entire input video.

Due to the temporal information inconsistency, the feature importance of each frame is different. Within the video, the frames with severe occlusions or misalignment problems may be less important than the other frames with a clear representation of the whole human body for the re-id task. Thus, it is intuitive to use the temporal attention to assign a weight to each frame to represent its feature importance in the entire video. Furthermore,

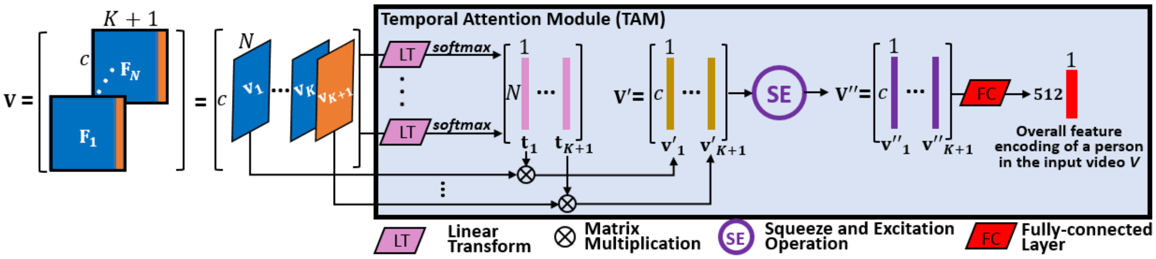


Figure 3.5. The computational graph of the Temporal Attention Module (TAM).

using a single weight to represent the feature importance of all body parts in each frame is inadequately robust[3], since those frames with serious occlusions may also contain some clear representations of a few body parts. Therefore, for each body part (or the whole body) in the video, our proposed Temporal Attention Module (TAM) assigns per-frame attention scores to represent its feature importance in different frames of the video.

Recall that in Eq. 3.10, the image feature of frame I_n in the video V is represented by $\mathbf{F}_n = [\mathbf{f}^{in}_{1,n}, ..., \mathbf{f}^{in}_{K,n}, \mathbf{f}^{out}_{n}] \in \mathbb{R}^{c\times(K+1)}$. For simplicity, we rewrite $\mathbf{F}_n$ as $\mathbf{F}_n = [\mathbf{f}_{1,n}, ..., \mathbf{f}_{(K+1),n}] \in \mathbb{R}^{c\times(K+1)}$, where $\mathbf{f}_{(K+1),n} = \mathbf{f}^{out}_{n}$. By combining image features from N frames in the video, we can define the video representation $\mathbf{V}$ as

$$\mathbf{V} = [\mathbf{F}_1, ..., \mathbf{F}_N], \tag{3.12}$$

where $\mathbf{V} \in \mathbb{R}^{c\times(K+1)\times N}$. After shifting the dimension, we can rewrite $\mathbf{V}$ as:

$$\mathbf{V} = [\mathbf{v}_1, ..., \mathbf{v}_{(K+1)}], \tag{3.13}$$

where $\mathbf{V} \in \mathbb{R}^{c\times N\times(K+1)}$, and $\mathbf{v}_k \in \mathbb{R}^{c\times N}$, $k \in [1, K+1]$, represents the feature representation of a body part (or the whole body), detected by the TSPAM in the entire video.

[3] We carry out experiments to compare assigning single per-frame weight and multiple per-frame weights in the *Analysis on Temporal Attention Module* section of the *Experiment* section.

To apply temporal attention to the k-th element $\mathbf{v}_k$ in $\mathbf{V}$, we learn attention weight vectors $\mathbf{w}^{(v_k)} \in \mathbb{R}^{c\times 1}$ and $\mathbf{b}^{(v_k)} \in \mathbb{R}^{N\times 1}$, and compute a temporal attention score vector $\mathbf{t}_k$, indicating the importance of $\mathbf{v}_k$ in different frames:

$$\mathbf{t}_k = (\mathbf{v}_k)^\top \mathbf{w}^{(v_k)} + \mathbf{b}^{(v_k)}, \quad \text{where} \quad \mathbf{t}_k \in \mathbb{R}^{N\times 1}. \tag{3.14}$$

Then, the temporal attention score vector $\mathbf{t}_k$ is passed through a *softmax* layer to get $\mathbf{t}_{k,softmax}$. Finally, we compute the attentive feature $\mathbf{v}'_k$ by

$$\mathbf{v}'_k = \mathbf{v}_k \mathbf{t}_{k,softmax}, \quad \text{where} \quad \mathbf{v}'_k \in \mathbb{R}^{c\times 1}. \tag{3.15}$$

Now, the input video can be represented by $\mathbf{V}' = [\mathbf{v}'_1, ..., \mathbf{v}'_{(K+1)}] \in \mathbb{R}^{c\times(K+1)}$, where $\mathbf{v}'_k$ is a $c \times 1$ feature vector that represents one specific body part (or the whole body if $k = K + 1$) over the entire video.

As the contributions of different body parts (or the whole body) may be different for the re-id task, inspired by the SENet Hu *et al.* (2018), we propose to apply a “Squeeze and Excitation” operation on $\mathbf{V}'$ to recalibrate each $\mathbf{v}'_k$ in $\mathbf{V}'$, according to the contribution of the k-th body part (the whole body if $k = K + 1$) to the re-id task. Specifically, we introduce a statistic $\mathbf{u} = [u_1, ..., u_{(K+1)}] \in \mathbb{R}^{1\times(K+1)}$, whose k-th element u_k is a scalar generated by squeezing $\mathbf{v}'_k$:

$$u_k = \frac{1}{c}\sum_{j=1}^{c} \mathbf{v}'_{j,k}. \tag{3.16}$$

Then we pass $\mathbf{u}$ through a linear transformation followed by a *sigmoid* activation:

$$\bar{\mathbf{u}} = \sigma(\mathbf{w}^{(u)} \circ \mathbf{u}), \tag{3.17}$$

where $\mathbf{w}^{(u)} \in \mathbb{R}^{1\times(K+1)}$ is the learnable parameter, σ denotes the *sigmoid* function, and $\bar{\mathbf{u}} = [\bar{u}_1, ..., \bar{u}_{(K+1)}] \in \mathbb{R}^{1\times(K+1)}$ is the excitation operator that represents the contribution importance of each body part (or the whole body) in the video for the re-id task. Finally, we recalibrate the k-th element v'_k in V' by

$$\mathbf{v}''_k = \bar{u}_k \mathbf{v}'_k, \quad \text{where} \quad \mathbf{v}''_k \in \mathbb{R}^{c\times 1}. \tag{3.18}$$

After the "Squeeze and Excitation" operation, the input video can be represented by a final attentive feature representation, $\mathbf{V}'' = [\mathbf{v}''_1, ..., \mathbf{v}''_{(K+1)}] \in \mathbb{R}^{c\times(K+1)}$, which will be flattened and fed into a fully-connected layer that represents the overall feature encoding of the input video on a person. The aggregated feature vector after the fully-connected layer is embedded into 512-dimensions.

3.3.7. Final Loss Function. We design the following loss to train our complete network (ResNet50-SAU-TSPAM-TAM) in an end-to-end fusion:

$$Loss_{final} = Loss_{hard\_triplet} + Loss_{cross\_entropy} + \lambda Reg_{cdiver}, \tag{3.19}$$

where λ is a hyperparameter to control the influence of the regularization term Reg_{cdiver} introduced in Eq. 3.11.

For the batch-hard triplet loss Hermans *et al.* (2017), we form a mini-batch by randomly sampling 48 identities, each of which has 6 randomly sampled video clips. For the cross-entropy loss, we pass the overall feature encoding of each video in a mini-batch through a 48-way fully-connected layer followed by a *softmax* layer.

3.4. EXPERIMENT

In this section, we validate the effectiveness of our framework in solving the temporal information inconsistency challenge on three challenging datasets.

3.4.1. Datasets and Evaluation Metrics. The proposed approach is evaluated on three challenging video-based person re-identification datasets: PRID2011 Hirzer *et al.* (2011), iLIDS-VID Wang *et al.* (2014), and MARS Zheng *et al.* (2016). PRID2011 consists of person videos of 200 identities captured by 2 cameras. iLIDS-VID consists of 600 videos of 300 identities captured by 2 cameras. MARS is the largest video-based person re-identification benchmark that consists of 20,715 videos of 1,261 identities generated by person-detectors. Each identity is captured by at least 2 cameras (up to 6 cameras).

We conduct experiments on PRID2011 and iLIDS-VID following the evaluation protocol from Wang *et al.* (2014) and report rank-1 accuracy. For the experiments on MARS, we follow the evaluation protocol provided by MARS and report rank-1 accuracy and mean average precision (mAP).

3.4.2. Implementation Details. Since the original person videos are of various-length, we divide each video into non-overlapping video clips, each of which contains 10 consecutive frames for re-id ($N = 10$). Each frame of the video clip is resized to 256×128. We augment the occlusion in the video data using Cut-out DeVries and Taylor (2017) with a 32×32 black region that randomly applies to the video frames. We train the proposed framework in an end-to-end fusion. 10% video clips are sampled from the training set as the validation set and the rest are for training the network. The network is updated using batched Stochastic Gradient Descent with an initial learning rate set to 0.05 which is dropped every 30 epochs by multiplying a decay rate 0.5 for 120 epochs.

3.4.3. Component Analysis. We conduct analytic experiments to investigate the effectiveness of each component of our framework. The experiment results are listed in Table 3.1.

3.4.3.1. Analysis on Semantic Attention Unit. *Baseline* corresponds to ResNet50. *Baseline-raw-mid-high* corresponds to the ResNet50 that concatenates the raw mid-level and high-level features as the final feature representation of each frame in the input video. *Baseline-SAU* corresponds to the ResNet50 with SAU embedded for bringing high-level semantics to mid-level features. These three models are trained with hard mining triplet loss and cross entropy loss. As shown in Table 3.1, *Baseline-raw-mid-high* outperforms *Baseline* due to the supplemental regional information introduced by mid-level features.

Compared with *Baseline-raw-mid-high*, *Baseline-SAU* improves the rank-1 accuracy by 1.8% on average, which shows that SAU is effective at suppressing irrelevant features and benefiting the re-id task.

3.4.3.2. Analysis on Two-level Spatial Attention Module. *Spattn-out* uses the output of *res5* in ResNet50 as the high-level feature from which the *zoom-out attention* captures the whole human body. *Spattn-in* uses the output of *res4* in ResNet50 with SAU embedded as the mid-level feature from which *zoom-in attentions* capture *K* discriminative body parts). *Spattn-in&out* has the same network architecture as *Spattn-in* and uses both *K zoom-in attentions* and the *zoom-out attention*. *Spattn-in&out+Reg* uses the complete Two-level Spatial Attention Module (TSPAM) which has the same network architecture as *Spattn-in&out* but with the constrained diversity regularization. For these four networks, image features extracted from individual frames are averaged over all frames without temporal attention and then sent to the last FC layer for the network training. From Table 3.1, *Spattn-in&out+Reg* obtains the best performance among these four networks. By discovering discriminative body parts with minimal overlaps from mid-level features as well as efficiently associating all body parts to the target person, the TSPAM is useful for the video-based re-id task.

3.4.3.3. Analysis on Temporal Attention Module. *Spattn-in&out+Reg+single* corresponds to the network using the complete TSPAM and temporal attention to assign a single temporal attention weight to each frame. *Spattn-in&out+Reg+multi* corresponds to the network using the complete TSPAM and temporal attention to assign multiple temporal attention weights to multiple body parts and the whole body in each frame without the "Squeeze and Excitation" operation. *Spattn-in&out+Reg+TAM* corresponds to our complete framework, which obtains the best results on three datasets in Table 3.1. This shows the effectiveness of the TAM in solving the *temporal information inconsistency challenge* in the video-based re-id.

Table 3.1. Component analysis on our approach on three datasets: PRID2011, iLIDS-VID and MARS.

	PRID2011	**iLIDS-VID**	**MARS**
Baseline	83.0%	61.2%	74.5%
Baseline-raw-mid-high	84.8%	64.9%	76.1%
Baseline-SAU	86.2%	67.8%	77.2%
Spattn-out	86.9%	69.3%	78.8%
Spattn-in	87.8%	71.2%	79.3%
Spattn-in&out	90.1%	72.9%	82.7%
Spattn-in&out+Reg	92.6%	75.7%	84.6%
Spattn-in&out+Reg+single	94.3%	78.3%	86.1%
Spattn-in&out+Reg+multi	96.8%	81.6%	87.9%
Spattn-in&out+Reg+TAM (Ours)	**97.3%**	**83.1%**	**90.2%**
Ours-w/-Reg-term2	96.3%	82.0%	89.1%
Ours-w/o-SAU	95.5%	81.0%	87.1%
Ours-w/o-SAU-w/-SAGU	96.8%	82.5%	88.6%

3.4.3.4. More analysis on Diversity Regularization and Semantic Attention Unit. *Ours-w/-Reg-term2* corresponds to our framework that only uses the second term of the proposed constrained diversity regularization in Eq. 3.11, which used to be introduced in Li *et al.* (2018a); Lin *et al.* (2017). Our framework outperforms *Ours-w/-Reg-term2*, which shows the importance of forcing different body parts (*zoom-in attentions*) to associate with the target person's whole body (*zoom-out attention*) other than blindly diversifying body parts without any constraint. *Ours-w/o-SAU* corresponds to our framework without SAU. *Ours-w/o-SAU-w/-SAGU* corresponds to our framework that replaces SAU with the Semantic Attention Guidance Unit proposed in Section 2 for using the high-level feature to guide the mid-level feature. Our framework outperforms *Ours-w/o-SAU* and *Ours-w/o-SAU-w/-SAGU*, which shows that the proposed SAU is effective at guiding discriminative mid-level features with high-level semantics to benefit the video-based re-id task.

Table 3.2. Analyzing the block of ResNet50 from which we extract mid-level features and the parameter *K* (the number of *zoom-in attentions*), on the validation dataset.

	PRID2011	**iLIDS-VID**	**MARS**
res2	95.5%	81.4%	83.2%
res3	96.2%	81.7%	87.7%
res4	**97.3%**	**83.1%**	**90.2%**
K = 1	95.3%	77.0%	86.2%
K = 3	96.0%	77.4%	87.3%
K = 5	**97.3%**	**83.1%**	**90.2%**
K = 7	96.9%	82.6%	89.7%

3.4.3.5. Analysis on different mid-level features. We also study the effect of changing the block of ResNet50 which we extract mid-level features from. The experiment results are listed in the upper part of Table 3.2. From the results, we can see that the features extracted from *res4* result in better performances than the features obtained from other layers. We believe that the lower-level information (e.g., outputs of *res2* or *res3*) is based on a small Effective Receptive Field (ERF) that overemphasizes the local details and lacks not only the semantics of the entire image but also regional component understanding of the human body. Compared with the outputs of other layers, the output from *res4* has a reasonable ERF that can capture salient body parts, which can be used as the regional information for the re-id task.

3.4.3.6. Analysis on the quantity of zoom-in attentions. The effect of varying the number *K* of *zoom-in attentions* is also studied. As shown in the lower part of Table 3.2, we can see the re-id performances are increasing when *K* is increased from 1 to 5. However, the performances slightly drop when *K* is increased from 5 to 7. In the re-id task, the target person only has a few general regions that contribute to the re-identification algorithms

Table 3.3. Comparison with the state-of-the-art methods on three challenging datasets.

	PRID2011	iLIDS-VID	MARS
RFA-Net Yan *et al.* (2016)	58.2%	49.3%	-
RNN McLaughlin *et al.* (2016a)	70.0%	58.0%	-
MARS Zheng *et al.* (2016)	77.3%	53.0%	68.3% (49.3%)
ASTPN Xu *et al.* (2017)	62.0%	77.0%	-
AMOC+EpicFlow Liu *et al.* (2017)	83.7%	68.7%	68.3% (52.9%)
PAM Khan and Bremond (2017)	92.5%	80.1%	-
MSML Xiao *et al.* (2017)	-	-	84.2% (74.6%)
DR+ST Li *et al.* (2018a)	93.2%	80.2%	82.5% (65.8%)
XQDA Zeng *et al.* (2018)	95.7%	80.5%	86.4% (79.3%)
STA Fu *et al.* (2019)	-	-	86.3% (80.8%)
VRSTC△ Hou *et al.* (2019)	-	83.4%	88.5% (82.3%)
AFDTA□ Zhao *et al.* (2019)	93.9%	**86.3%**	87.0% (78.2%)
Ours	**97.3%**	83.1%	**90.2% (82.2%)**

(e.g., head, arms, legs, etc.). Distributing the visual attention on too many regions will distract the network from focusing on major component changes of the target person, which leads to inferior performance. So, we use $K = 5$ in our experiments.

3.4.4. Comparison with the State-of-the-art Models. We report the performance comparison between our approach and other state-of-the-art methods on three widely-used video re-id datasets in Table 3.3. Our framework improves the best reported re-id performances on PRID2011 and MARS by 1.6% and 1.7% in the rank-1 accuracy, respectively. On iLIDS-VID, our framework obtains slightly lower performances than VRSTC Hou *et al.* (2019) and AFDTA Zhao *et al.* (2019), however, which require extra synthetic human body part datasets (marked with △ in Table 3.3) and extra labels of human appearance attributes (marked with □ in Table 3.3) besides person IDs, respectively. Another state-of-the-art, DR+ST Li *et al.* (2018a), which also proposes a spatial and temporal attention network that is pretrained on extra image-based person re-id datasets to discover different body parts from high-level features for the re-id task. Compared with these methods, our framework is trained in an end-to-end fashion without requiring extra

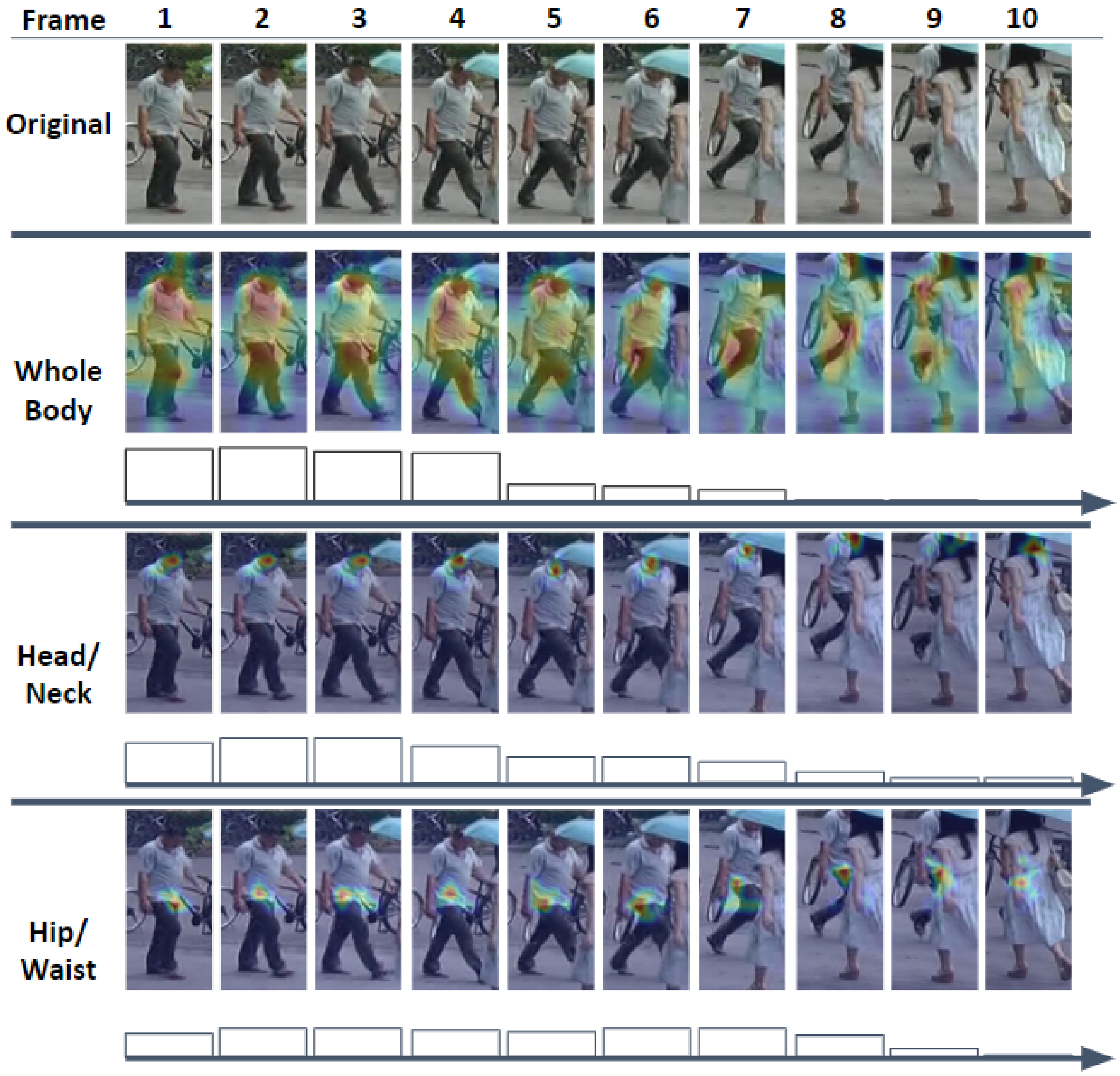


Figure 3.6. Qualitative results of our proposed network.

image-based datasets, labels, or pretraining steps, and it can highlight the target person's whole human body from the high-level feature and capture his/her different discriminative body parts from the mid-level feature over time, which is effective for solving the temporal information inconsistency challenge in the video-based re-id task. Some qualitative results of our framework are shown in Figure 3.6.

3.5. SUMMARY

Motivated by the unanswered research question of Section 2 and its challenging extension in natural image sequences, we propose a novel attention framework to handle this challenge in the video-based person re-identification task. In this work, we design a Semantic Attention Unit to extract discriminative mid-level features with selective high-level object semantic information to describe body parts, a Two-level Spatial Attention Module and a Temporal Attention Module to learn an overall semantic feature representation of the target person in a video, which represents different body parts and the whole visible body using mid- and high-level features, respectively, over the duration of the entire video. The proposed method is extensively evaluated on three challenging video-based person re-id datasets and shows competitive performance on the video-based re-id task.

3.6. FUTURE WORK

Although the proposed attention framework (SAU-TSPAM-TAM) is effective in extracting all useful semantic information of the target person in the video for the challenging video-based person re-identification task, we notice that our newly proposed attention mechanisms are not directly supervised. For instance, in our TSPAM, all the spatial attention models for discovering the whole visible human body and his/her body parts are supervised by the final loss function (shown in Eq. 3.19) for the re-identification task, instead of a loss function that assesses the performance of discovering the human body and body parts. Current best strategy to address this issue is to pre-train an attribute classification model by using extra appearance attribute labels to supervise the spatial attention models Lin *et al.* (2019); Su *et al.* (2016). However, the main drawback of this strategy is that the extra attribute labels are expensive, error prone, and highly subjective to different labelers. For instance, in Lin *et al.* (2019), the proposed attribute clasification model requires 27 hand-annotated attributes (e.g., gender, upper-body length, clothe colors, etc.) on over

500,000 person images for training to perform accurate atttribute classication. The second best strategy is to collect human viusal attentions on the image to guide the attention model to focus specific image regions for different computer vision tasks Linsley *et al.* (2018). However, this strategy is also requring expensive annotation efforts. For instance, in Linsley *et al.* (2018), the author collects over 400,000 attention heatmaps on 196,499 unique images from 1,235 labelers. Therefore, the motivated research question is: *how to supervise the attention model without requiring extra annotations?* We will try to address this challenge by introducing mutual learning to multiple correlated attention models in the future work.

4. CONCLUSION

Traditional attention mechanism has been widely used in the computer vision community for helping Deep Neural Networks to learn semantic meanings of the input data for different computer vision tasks. In this study, we take the traditional attention mechanism a step further by proposing four attention models that can be embedded into different DNNs to conduct semantic attention guided feature fusion, correlative multi-level feature fusion, multiple visual cues discovering, and temporal information selection, respectively. Specifically, in Section 2, we propose a deep active learning framework that consists of: (1) a semantic attention guided fully convolutional network (sag-FCN) embedded with multiple Semantic Guidance Attention Units, which can automatically highlight salient features of the target content for accurate pixel-wise predictions and (2) a distribution discrepancy based active learning algorithm that progressively suggests valuable samples to train the sag-FCNs.

In Section 3, we propose a novel attention framework to handle this challenge in the video-based person re-identification task. In this work, we design a Semantic Attention Unit to extract discriminative mid-level features with selective high-level object semantic information to describe body parts, a Two-level Spatial Attention Module, and a Temporal Attention Module to learn an overall semantic feature representation of the target person in a video, which represents different body parts and the whole visible body using mid- and high-level features, respectively, throughout the entire video.

In closing, we conclude that the major obstacle of the development of attention mechanism (AM): AM is normally supervised by a loss function for a specific task instead of being supervised by a loss function that assesses the accuracy of AM, which might produce inaccurate attention maps and finally cause inferior performances. Existing solutions to this challenge always require a large amount of extra untransferable annotations, which is impractical for extended research or real-world applications. In future work, we will address this challenge by introducing mutual learning to multiple correlated attention models to obtain accurate attention maps without requiring extra annotated data.

APPENDIX A.

EARLY DROUGHT PLANT STRESS DETECTION WITH BI-DIRECTIONAL LONG-TERM MEMORY NETWORKS

1. INTRODUCTION

On a global basis, drought, in conjunction with high temperature and radiation, poses the most important environmental constraint to plant survival and to crop productivity Boyer (1982). Agriculture is the major victim of drought in many regions of the world. Because the usable water supply in the world is limiting, the future food demand for rapidly increasing population pressures will be further aggravating the effects of drought Somerville and Briscoe (2001), which calls for attention to advance research to improve the breeding strategies of drought tolerant plants and early drought stress detection approaches.

2. RELATED WORK

The mechanism of drought tolerance in plants has been discussed at the molecular level Hasegawa *et al.* (2000). There are three main mechanisms in drought plants which reduce the crop yield: (i) reduced canopy absorption of photosynthetically active radiation, (ii) decreased radiation-use efficiency, and (iii) reduced harvest index Earl and Davis (2003). However, the reproducibility of drought stress treatments to these mechanisms is cumbersome, which has hindered both traditional breeding efforts and modern genetic approaches in the improvement of drought tolerance of crop plants Xiong *et al.* (2006). In addition, the mechanistic basis underlying drought tolerance is complex as it is mainly contributed by related traits that are mostly determined by polygenic inheritance Römer *et al.* (2012). In recent years, by measuring the structural and functional status of plants, phenomic approaches may overcome the limited predictability. However, the lack of high throughput phenomic data has been labeled as the “phenomic bottleneck” Richards *et al.* (2010).

In the past years, as imaging systems and image analysis techniques are developing, hyperspectral cameras have been widely used in plant science research, such as monitoring the growing condition of crops. In hyperspectral imaging, the measured radiative properties of plant leaves or canopies can be used to determine structural and physiological traits of vegetation Malenovskỳ *et al.* (2009); Ustin and Gamon (2010), for instance, a low reflectivity in the visible part of the spectrum can be used to characterize the spectral reflectance as a strong absorption by photosynthetic pigments, whereas a high reflectivity in the near infrared is produced by a high scattering of light by the mesophyll tissues of the leaf. In addition, the reflectivity in the shortwave infrared part of the spectrum is determined by the water, protein, cellulose, and lignin content of plant tissues Rascher *et al.* (2010). However, the spectral reflectance is a combination of multiple physiological traits. Despite several laboratory studies that have shown a relationship between the amount of water in the leaf and the reflectance intensity in the short infrared part of the spectrum, the determination of the water content presents some difficulties, due to the large reflectance variation among leaves with the same water status. The most challenging issue in estimating the water content using the spectral reflectance information is the decoupling of the contributions of water content and other physiological traits.

Remote sensing has been successfully used in the precision agriculture for providing the timely crop condition information during growing seasons. In the optical region, the vegetation indices (VIs) have been used to detect crop conditions, such as the water content and the nitrogen status. Most approaches are aiming at quantifying plant traits by calculating VIs that quantify specific plant structural changes Fiorani *et al.* (2012). Although VIs have been widely used to detect multiple crop growing stresses in the advanced stage, such as the leaf nitrogen and the chlorophyll content Haboudane *et al.* (2008); Tilling *et al.* (2007), the crop biomass Thenkabail *et al.* (2000) and the vegetation moisture content Yilmaz *et al.* (2008), the use of VIs for the early drought stress detection is still challenging, because different crop stresses have similar VI computations in the early stage Römer *et al.* (2012). Furthermore, the high cost of the hyperspectral camera system and its further maintenance is limiting the development of drought stress detection approaches based on the hyperspectral image analysis for the consumer applications.

2.1. MOTIVATION AND CONTRIBUTION

From the computer vision perspective, the image analysis based drought stress detection can be defined as the classification of images containing drought plants or not. Most previous approaches aimed to extract specific handcrafted features (e.g., spectral reflectivity and vegetation indices (VIs)) from hyperspectral images to recognize plants under the drought condition. The main problem of these methods is the feature selection. Spectral reflectivity, VIs and other indices, which have been widely used as features in the previous studies Malenovskỳ *et al.* (2009); Römer *et al.* (2012), are pixelwise calculations of pixel intensities in the individual hyperspectral image. In other words, they are all pixel- wise features that represent crop growing conditions at one time instant. However, as the plant is growing, the drought stress condition should be a continuous procedure with a unique time-series variation pattern, which can be well represented by temporal features. Thus, compared to the individual image, a time-series image sequence containing the temporal information will be a better representation of the plant grow ing condition for the early drought stress detection task.

Several previous studies have shown a relationship between the leaf water stress and the spectral reflectance variation in the visible region Danson *et al.* (1992); Hunt Jr and Rock (1989). This investigation indicates that the RGB image is able to be used for the early drought stress detection task. In other words, considering the temporal features, the early drought stress detection problem can be formulated as the classification of time-series RGB image sequences containing drought plants or not. Compared to previous approaches using hyperspectral images, the methods based on RGB image analysis are more cost-effective for the consumer applications.

In recent years, long short-term memory (LSTM) networks have been widely used in different real world classification tasks, such as action recognition Baccouche *et al.* (2011); Liu *et al.* (2016), event detection Feng *et al.* (2018); Parascandolo *et al.* (2016) and natural language processing Wang and Jiang (2015); Wen *et al.* (2015). As an effective method to uncover the hidden temporal relation in time-lapse data and classify the sequential data, LSTM is suitable for our task of early drought stress detection.

In this study we proposed a Bidirectional Long Short-Term Memory (BLSTM) model to solve the problem of early drought plant stress detection on RGB images. First, we extract the time-series image patch sequences that contain the temporal variation information of the plant as patch sequences. Secondly, a pre-trained Convolutional Neural Network model is used for extracting discriminative features from each image in the patch sequences.

Finally, the patch sequence, in the form of a sequence of feature vectors, will be input to the BLSTM for the binary classification. Two independently collected datasets are used to validate the performance of our proposed method.

The main contributions of this work are: (i) the application of BLSTM to RGB image sequences for early drought plant stress detection for the first time, (ii) the investigation of the earliest moment that we can detect the plant drought stress condition from RGB images, and (iii) the proposal of an efficient RGB image data collection strategy that can use less time and manpower for the purpose of accurate early drought plant stress detection.

The rest of this paper is organized as follows: in the next section, we introduce the methodology of our proposed method including data acquisition, data preparation, the proposed BLSTM model, and the proposal of an efficient RGB image data collection strategy; Next, we discuss the data collection guidance and validate our method on two RGB image datasets followed by our paper's conclusions.

3. METHODOLOGY

In this section, we first introduce the two plant image datasets tested in this work; then, illustrate the pipeline of our proposed method, including the data preparation step and the BLSTM model. Finally, we introduce the RGB image collection strategy for the drought plant stress detection.

3.1. DATA ACQUISITION

Two independently collected RGB image datasets of crop plants are utilized in this work. The *LemnaTecDD Dataset* is collected from the LemnaTec platform Virlet *et al.* (2017) at the Donald Danforth Plant Science Center. In the LemnaTec platform, plants are automatically transported by conveyers through a series of imaging cabinets to capture images from two sides, as well as, from the above. To collect this dataset, 10 replicates of 27 nested association mapping[1] (NAM) lines of maize are planted in a randomized complete block design with four different watering regimes (25% FC[2] , 50% FC , 75% FC and 100% FC). Starting at the 15th day after planting (DAP), plants are imaged daily for ten days (15th to 24th DAP) (image samples are shown in Figure A3.1). The image resolution is $2,454 \times 2,056$ pixels. The experiment is repeated four times. Therefore, there are 1,080 $(10 \times 27 \times 4)$ plants involved in this dataset. Each plant has three image sequences (two side view sequences and one top view sequence), each of which is considered as an independent plant sample. In this work, those image sequences containing plants with 25% FC , 75% FC or 50% FC watering regimes will be considered as the drought samples, and image sequences containing plants with 100% FC watering regimes will be the control samples.

[1] Nested Association Mapping (nam) is a technique designed for identifying and dissecting the genetic architecture of complex traits in corn Yu *et al.* (2008).

[2] FC, field capacity, is the amount of soil moisture or water content held in the soil after excess water has drained away and the rate of downward movement has decreased Israelsen and West (1922).

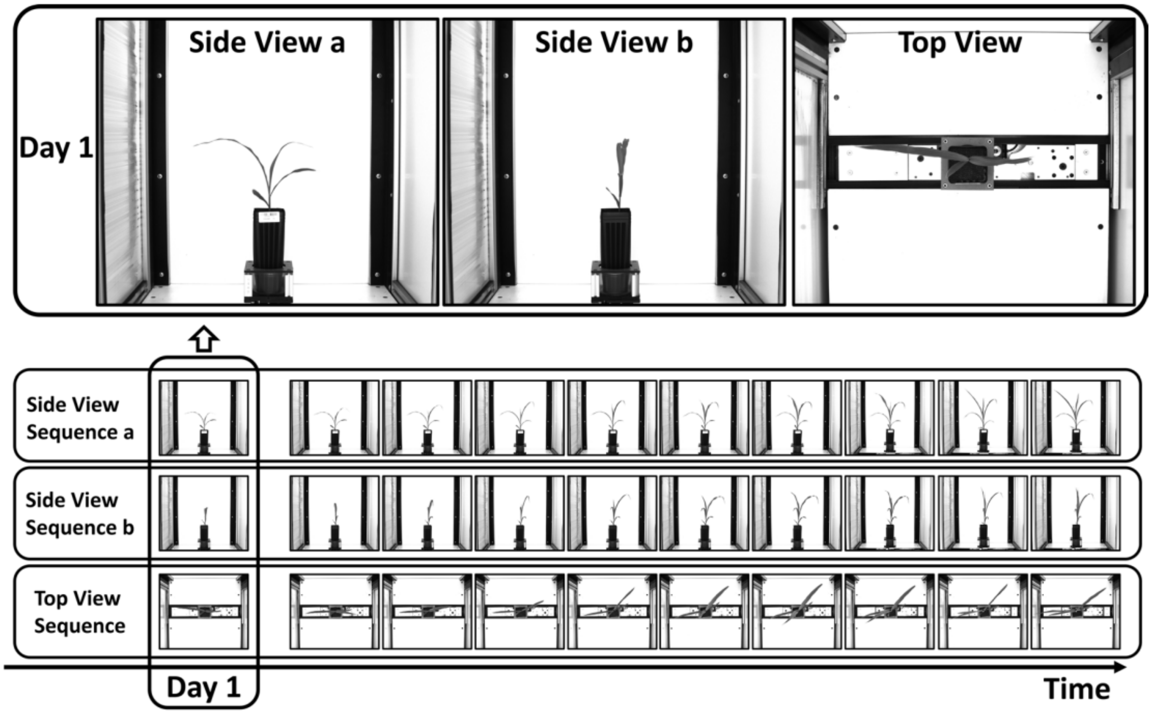


Figure A3.1. Image samples in the *LemnaTecDD Dataset*.

The *MSTCivil Dataset* is collected in the greenhouse of the Civil, Architectural, and Environmental Engineering Department in Missouri University of Science and Technology. In this dataset, there are three kinds of crops: drought tolerant maize, drought-susceptible maize, and sorghum. For each kind of crop, 16 replicates are planted, where 8 replicates are grown under the drought stress condition and the other 8 replicates are under the control condition. In the greenhouse, there are four RGB cameras installed on the ceiling to collect image data from the top (image samples are shown in Figure A3.2). Camera 1 and Camera 2 are for the drought groups. Camera 3 and Camera 4 are for the control groups. The image resolution is 1280 × 1040 pixels. Starting from the first day of planting, the four cameras image the plants hourly from 6 a.m. to 5 p.m. (12 hours) every day for 30 days. For all the images collected by a certain camera, those images captured at the same time instant will be grouped into an image sequence. For example, for all the images collected by Camera 1 in

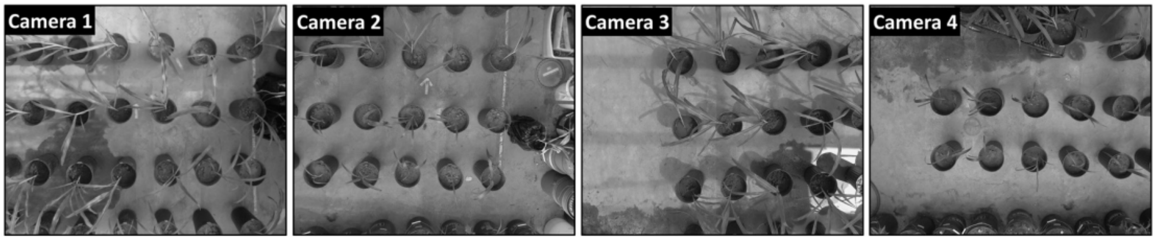


Figure A3.2. Image samples in the *MSTCivil Dataset*.

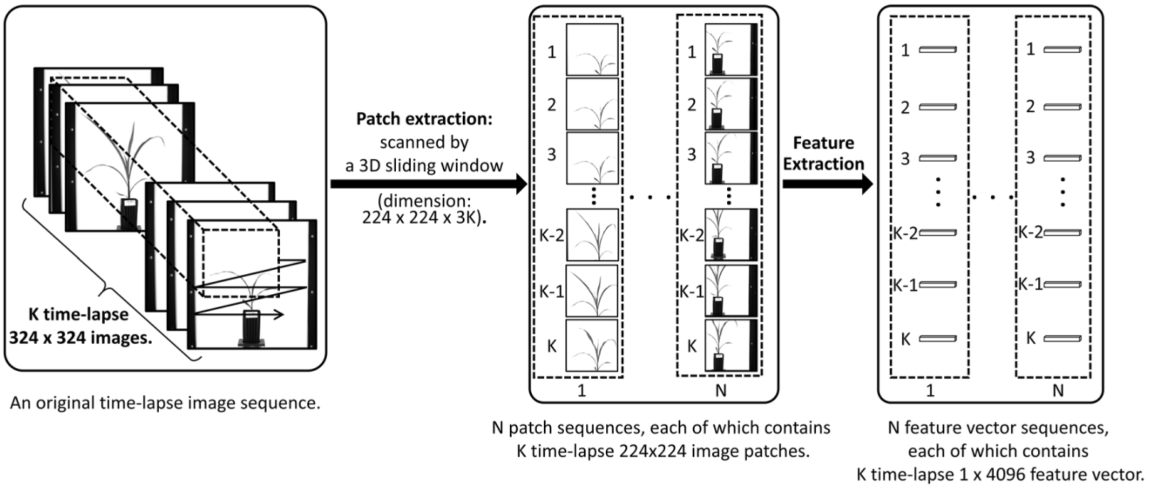

K time-lapse
324 x 324 images.
An original time-lapse image sequence.
Patch extraction:
scanned by
a 3D sliding window
(dimension:
224 x 224 x 3K).
1
2
3
K-2
K-1
K
1
N
N patch sequences, each of which contains
K time-lapse 224x224 image patches.
Feature
Extraction
N feature vector sequences,
each of which contains
K time-lapse 1 x 4096 feature vector.

Figure A3.3. The illustration of the data preparation process on the *LemnaTecDD Dataset*. The same data preparation process is also applied on the *MSTCivil Dataset*.

30 days, those images taken at 6 a.m. will be grouped into an image sequence that contains 30 time-lapse images. Therefore, in the *MSTCivil Dataset*, there are 48 (four cameras, hourly over 12 hours) image sequences, each of which contains 30 images over 30 days.

3.2. DATA PREPARATION

The data preparation process, as illustrated in Figure A3.3, includes two steps (extracting patch sequences from the image sequence and extracting feature descriptions from the images), which aims to transform the image data of plants into a form that is suitable for our proposed BLSTM model for the final classification for drought and non-drought plants.

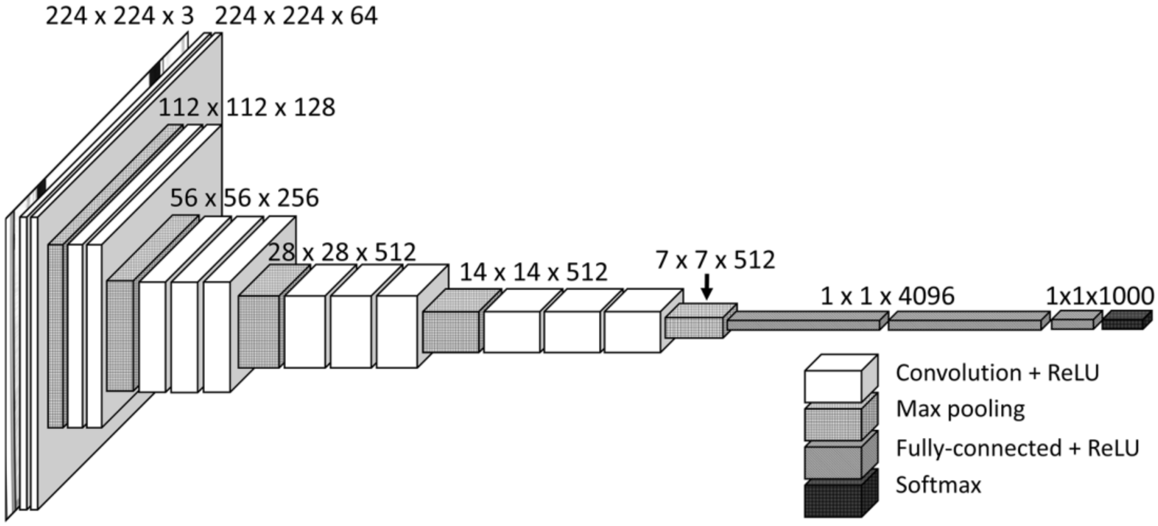


Figure A3.4. The architecture of the VGG-16 CNN model Simonyan and Zisserman (2014)

3.2.1. Patch Sequence Extraction. Given a plant sample in the form of a time-lapse image sequence that contains K time-lapse images, we first downsize the images to 324×324 pixels. Then, a 3D sliding window is applied on the downsized image sequence (dimension: $324 \times 324 \times 3K$[3]) to crop image patch sequences that contain a part of the plant. The size of the 3D sliding window is set as $224 \times 224 \times 3K$ with the stride size of 10 pixels. After the patch sequence extraction step, we obtain time-lapse patch sequences, each of which can be considered as a patch sequence for the classification.

In this study, instead of focusing on the entire plant, we are more interested in the temporal variation pattern in the patch sequence that only contains a part of the plant. Whereas the entire plant is able to provide the morphological information that helps detecting the drought plant, in practical cases, it is hard to obtain images taken by UAVs[4] containing the entire plant structure without any occlusion from a densely planted crop field.

[3] In the downsized image sequence, there are k RGB images, each of which can be represented by a $324 \times 324 \times 3$ matrix. Thus, the downsized image sequence can be represented as a $324 \times 324 \times 3K$ matrix, where each RGB image has three channels.

[4] UAV, unmanned aerial vehicle, commonly known as a drone, is an aircraft without a human pilot aboard.

Therefore, we decided to ignore the morphological information by using patch sequences. This strategy enables our proposed method to be applicable and robust in detecting the drought plant stress condition in challenging datasets, such as the *MSTCivil Dataset*.

3.2.2. Feature Extraction. After we obtain the patch sequences, instead of using handcrafted features, we use Convolutional Neural Networks (CNN) to extract discriminative features from each image in the patch sequence. In the previous works, handcrafted features have been widely used for representing crop stress conditions Rascher *et al.* (2010); Yilmaz *et al.* (2008). However, due to the complex physiological effects of the drought stress condition, handcrafted features, which are mostly focusing on specific characteristics, might discard significant amounts of the underlying conceptual information. The difference between any handcrafted features versus features learned by CNN is that multi-layered learning models, such as CNN, not only are able to explore low-level features from lower layers, but also can yield conceptual abstractions from higher layers. Hence, CNN is suitable for our feature extraction task.

In this work, the pre-trained CNN model, *VGG-16* Simonyan and Zisserman (2014) (shown in Figure A3.4), is adopted for the feature extraction. Based on the *VGG-16* model with weights pre-trained on ImageNet Deng *et al.* (2009), we first proceed to fine-tune the model for the drought plant classification task, and then we use the fine- tuned model as a feature extractor for our feature extraction task on the image patch sequences.

Fine-tuning a deep learning network is a procedure based on the concept of transfer learning Bengio (2012); Donahue *et al.* (2014). We first initialize the *VGG-16* model using the weights learned from the ImageNet dataset. Then, we truncate the last layer of the model, which is a softmax layer that targets at 1,000 classes of the ImageNet dataset, and replace it with a new softmax layer that targets at two classes (the drought condition and the control condition). The new softmax layer is trained using the backpropagation algorithm with our plant image data, which are the image patches in the patch sequences (the image patch will inherit the label from the patch sequence it belongs to). In order to transfer the

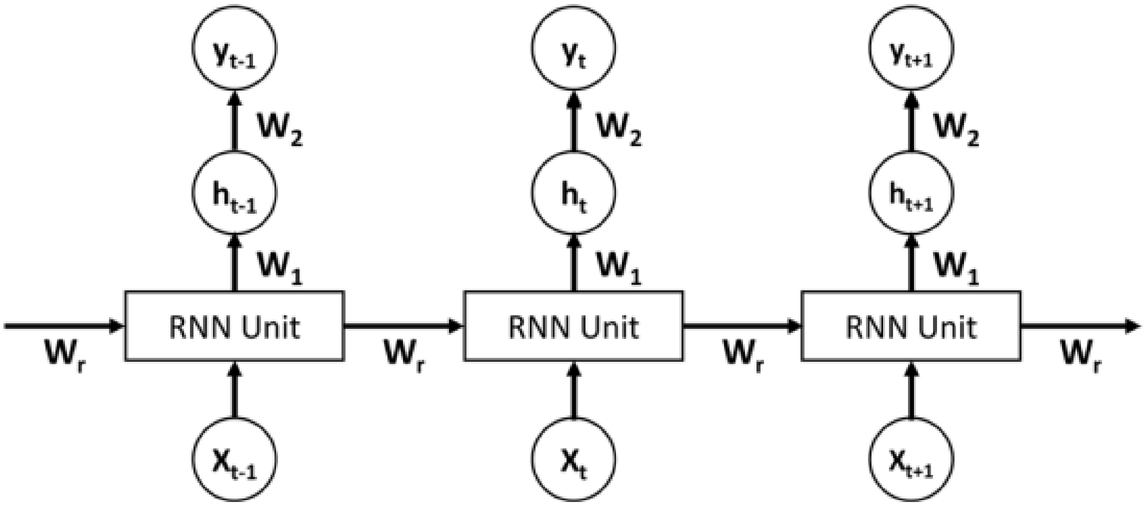


Figure A3.5. An example of recurrent neural network.

knowledge learned from the broad domain (ImageNet dataset) into our specific domain, we freeze the weights for the first ten layers so that they remain intact throughout the fine-tuning process. To fine-tune the model, we minimize the cross entropy function using the stochastic gradient descent algorithm with an initial learning rate of 10^{-4} , which is smaller than the learning rate for training the model from scratch. Finally, we use the fine-tuned model to extract spatial features from the image in the patch sequence. Each image in the patch sequence will be fed to the fine-tuned model as the input, which is passed through a stack of convolutional layers and three fully connected layers. The activation before the last fully connected layer will be considered as the extracted feature vector of the input. Thus, after the feature extraction step, the patch sequence will be represented by a sequence of feature vectors for the classification, where each image in the patch sequence will be represented by a 1×4096 feature vector in the feature vector sequence.

3.3. BIDIRECTIONAL LONG-SHORT TERM MEMORY RECURRENT NEURAL NETWORK

Recurrent Neural Network (RNN) is a layered neural network that uses its cyclic connection to learn temporal dependencies in the sequential data. The structure of a simple RNN is shown in Figure A3.5. Compared to feed-forward layered neural networks, for instance the multilayer perceptron SUTER (1990) that can only learn static pattern mappings, RNN can propagate prior time information forward to the current time for learning the context information in a sequence of feature vectors. In other words, the hidden layer of an RNN serves as a memory function.

An RNN can be described mathematically as follows. Suppose there is a sequence of feature vectors denoted as $x_t, t \in [1, T]$. In the RNN , the hidden layer output vector h_t and the output layer y_t are calculated as follows:

$$h_t = f(W_1 x_t + W_r x_{t-1} + b_1) \tag{A3.1}$$

$$y_t = g(W_2 h_t + b_2) \tag{A3.2}$$

where W_j and b_j represent the input weight matrix and bias vector of the $j^t h$ hidden layer, respectively, and W_r is denoted as a recurrent weight matrix; f and g represent activation functions of the hidden layer and output layer, respectively.

The traditional RNN, especially trained with gradient descent, has a significant problem called the vanishing gradient Hochreiter (1998). This problem causes traditional RNN to forget information after just a few steps. Long short-term memory (LSTM), a special kind of architecture RNN that remembers information for long periods of time, is designed to overcome the vanishing gradient problem Gers *et al.* (1999). In LSTM, the hidden cells of the traditional RNN are replaced by memory blocks. Therefore, LSTM is capable to find and exploit long range dependencies in the sequential data. Normally, each LSTM memory block consists of a memory cell and three gates: the input gate, the output

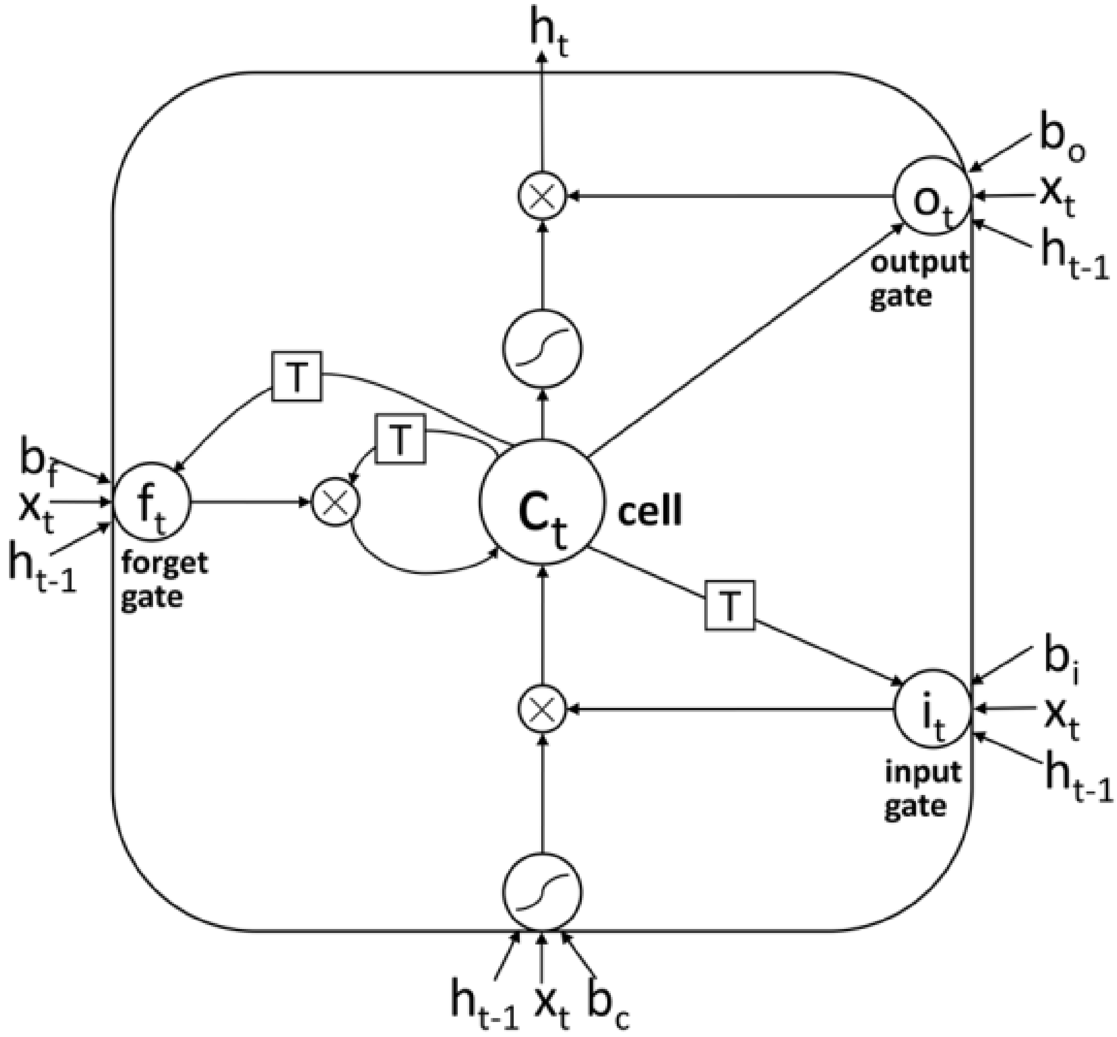


Figure A3.6. The structure of the LSTM block.

gate and the forget gate. These gates control the information flow in the LSTM block. As shown in Figure A3.6, the forget gate can reset the cell variable by forgetting the stored input c_t , while the input and output gates are in charge of reading input from x_t and writing output to h_t , respectively:

$$c_t = f_t \otimes c_{t-1} + i_t \otimes tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad \text{(A3.3)}$$

$$h_t = o_t \otimes tanh(c_t) \quad \text{(A3.4)}$$

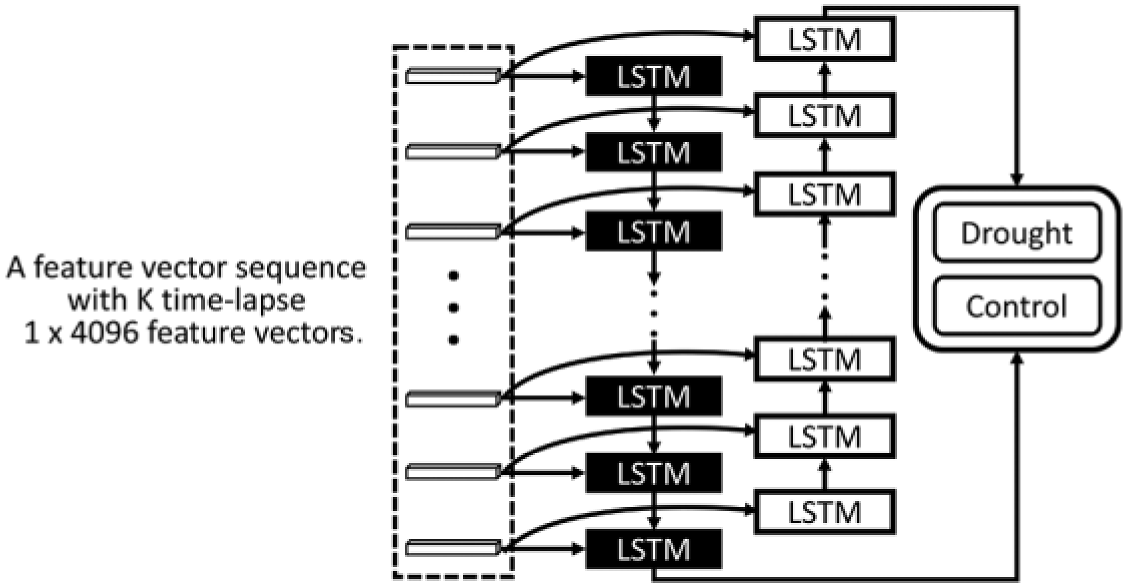

A feature vector sequence with K time-lapse 1 x 4096 feature vectors.
LSTM
Drought
Control

Figure A3.7. The architecture of the proposed BLSTM model that takes the time-lapse feature vector sequence as the input for the final classification. Each time-lapse feature vector sequence is representing a patch sequence.

where ⊗ denotes element-wise multiplication and *tanh* is the *hyperbolic tangent function* that is also applied in an element-wise manner. i_t , o_t and f_t are representing the output of the input gate, output gate and forget gate, respectively, while b_c is a bias term and W is the weight matrix. Since each LSTM memory block is an independent unit, the activation vectors i_t , o_t , f_t and c_t are all of same size as h_t . Note, each gate is only dependent on the cell within the same memory block.

To solve the early drought plant stress detection task, in addition to using LSTM memory blocks to remember information for a long period of time, we adopt the bidirectional mechanism from the bidirectional RNN (BRNN) Schuster and Paliwal (1997) to design a bidirectional LSTM (BLSTM) model for processing the input sequential data in both temporal directions. The motivation of bringing the bidirectional mechanism to our scheme is to enable our proposed model to have a better understanding of the unique variation pattern of drought plants. Because the water content variation of the plant in the early stage is subtle to be recognized, especially for those plants with the mild drought stress, such as the plants with 50% FC or 75% FC watering regimes in the *LemnaTecDD Dataset*. Compared with the LSTM model that can only process input data in one direction, the proposed BLSTM model, which can process input data in forward direction and backward direction respectively, is capable of exploring the full context information of the water content variation for identifying subtle differences between drought plants and normal plants.

The proposed BLSTM model is depicted in Figure A3.7. In the input-layer, each hidden layer has K LSTM blocks, each of which takes a feature vector in the feature vector sequence as the input. In LSTM blocks, gates are activated using the standard sigmoid function, $(1 + e^{˘x})^{˘1}$, and the block input as well as the block output is squashed with the *hyperbolic tangent function* (*tanh*). After presenting an input sequence entirely to the BLSTM , the result can be read at the output-layer. The time-lapse feature vector sequence, which is the representation of a patch sequence, will be classified as either the drought condition or the control condition.

To train the BLSTM model, feature vector sequences extracted from patch sequences containing drought plants are considered as positive samples (the drought condition), while feature vector sequences extracted from patch sequences containing control plants are negative samples (the control condition). Both of these two kinds of samples will be fed to the model. The proposed BLSTM model is built using Python on the Keras Chollet (2015) toolkit with a Tensorflow backend. The weights of the network are initialized from a $N(0.0, .05)$ distribution, while we add 1 to the forget gate bias of LSTMs at initialization. The network is trained with an RMSprop optimizer, where the learning rate is $10^{˘3}$ with the decay rate as $10^{˘6}$. The applied dropout probability in our network is 0.5. We train the model for 50 epochs and use early stopping based on the validation performance.

3.4. OPTIMAL DATA COLLECTION STRATEGY

In addition to the BLSTM model for the drought plant stress detection, we first propose the Optimal Data Collection Strategy (ODCS) that can use less information to accurately detect the plant drought stress from RGB images as early as possible. The image data collection for the drought stress detection task is expensive, especially in crop fields. In order to acquire detailed variation information of the growing plants, we have to collect the image data from the first few days after planting as frequently as possible, which is inefficient. To eliminate this issue, we propose the ODCS that aims to use as less image data as possible in a time period that is informative for the early drought plant stress detection task.

To find the ODCS , we design different data sampling strategies on the plant image dataset to simulate different data collection strategies. However, it is unpractical to try all possible sampling strategies, which will be a huge number of experiments testing the proposed method using different combinations of the images in the time-lapse patch sequence. Therefore, we select 15 representative data sampling strategies which can cover most of the time periods in the time-lapse patch sequence.

We intuitively design the sequence lengths of the data sampling strategies as 10 images, 15 images, and 20 images. The data sampling strategy is named as $S_mF_pL_q$, which means the original time-lapse patch sequence is sampled with time interval m from the p^{th} (First) image to the q^{th} (Last) image in the sequence. For instance, the $S_1F_11L_{30}$ strategy, which samples the original time-lapse patch sequence from the 11^{th} image to the 30^{th} image with time interval 1, is simulating the data collection strategy that collects plant image data every other day from the 11^{th} day to the 30^{th} day after planting. If the $S_1F_11L_{30}$ strategy can achieve competitive classification result on the early drought plant stress detection task, then, instead of collecting data every day for 30 days, we can use a lower data collection frequency by imaging the plant every other day during 20 days to save the manpower and the time. Note, in a data sampling strategy $S_m F_p L_q$, if the time interval $m = 0$, we will select the consecutive image patches from the p^{th} image to the q^{th} image in the original time-lapse patch sequence. These 15 sampling strategies are summarized in Table A4.1.

The proposed BLSTM model will be tested by using the sampled data sequences generated by different strategies. Correspondingly, in order to take input data in different lengths, the layout of the proposed BLSTM architecture will be adjusted by varying the number of LSTM blocks. The classification performances of these strategies will be compared and discussed to find the ODCS.

4. EXPERIMENTS

The main goals of this work are: (i) the application of the proposed BLSTM model to RGB images for the early drought plant stress detection task, (ii) the investigation of the earliest moment that we can accurately detect the drought plant stress condition from RGB images, and (iii) the proposal of an efficient RGB image data collection strategy that can reduce the amount of time and manpower and guarantee the accuracy of early drought plant stress detection at the same time.

To validate the first goal, the proposed BLSTM model is compared with the bidirectional RNN (BRNN) model, the LSTM model, and the CNN model. For the second and third goals, we design different data sampling strategies and compare their classification performances to find the Optimal Data Collection Strategy (ODCS).

In this section, we first describe evaluation metrics. Then, we compare different sampling strategies to find the ODCS. Finally, we validate the effectiveness of our proposed method on the *LemnaTecDD Dataset* and the *MSTCivil Dataset*, respectively.

4.1. EVALUATION METRICS

We adopt the leave-one-out policy in the experiment. In each dataset, the image data will be evenly separated into four subsets, where three subsets are used for training and the last one is for testing. We perform the leave-one-out experiment four times with each subset as the testing set alternatively. Then, the average performance on the four experiments in terms of precision, recall and F-score is utilized as the evaluation metrics.

In the experiments, we evaluate both the patch sequence classification and the image sequence classification. The patch sequence classification is the BLSTM model's prediction on the input patch sequence. In the data preparation step, by applying the 3D sliding window (dimension: $224 \times 224 \times 3K$) on the image sequence (dimension:

Table A4.1. The description of the 15 data sampling strategies. The data sampling strategy is named as $S_m F_p L_q$, which means that the data sequence is sampled with time interval m from the p^{th} (First) image to the q^{th} (Last) image in the sequence.

Sequence Length	**Name**	**Description**
10 images	$S_0F_1L_{10}$	Sampled with time interval 0 from 1^{st} image to 10^{th} image.
	$S_0F_{11}L_{20}$	Sampled with time interval 0 from 11^{th} image to 20^{th} image.
	$S_0F_{21}L_{30}$	Sampled with time interval 0 from 21^{st} image to 30^{th} image.
	$S_1F_1L_{20}$	Sampled with time interval 1 from 1^{st} image to 20^{th} image.
	$S_1F_6L_{25}$	Sampled with time interval 1 from 6^{th} image to 25^{th} image.
	$S_1F_{11}L_{30}$	Sampled with time interval 1 from 11^{th} image to 30^{th} image.
	$S_2F_1L_{30}$	Sampled with time interval 2 from 1^{st} image to 30^{th} image.
15 images	$S_0F_1L_{15}$	Sampled with time interval 0 from 1^{st} image to 15^{th} image.
	$S_0F_{11}L_{25}$	Sampled with time interval 0 from 11^{th} image to 25^{th} image.
	$S_0F_{16}L_{30}$	Sampled with time interval 0 from 16^{th} image to 30^{th} image.
	$S_1F_1L_{30}$	Sampled with time interval 1 from 1^{st} image to 30^{th} image.
20 images	$S_0F_1L_{20}$	Sampled with time interval 0 from 1^{st} image to 20^{th} image.
	$S_0F_6L_{25}$	Sampled with time interval 0 from 6^{th} image to 25^{th} image.
	$S_0F_{11}L_{30}$	**Sampled with time interval 0 from 11^{th} image to 30^{th} image.**
30 images	$S_0F_1L_{30}$	Sampled with time interval 0 from 1^{st} image to 30^{th} image.

$324 \times 324 \times 3K$) with the stride size of 10 pixels, the image sequence is decomposed into 100 patch sequences. Therefore, the image sequence classification is voted by its 100 decomposed patch sequences. For instance, if more than half of these patch sequences are classified as the drought condition, then the image sequence will be classified as the drought condition.

One of the comparison methods, the CNN method, takes the individual patch as the input and performs patch-wise classification rather than sequence-wise classification. Therefore, to evaluate the patch sequence classification of the CNN method, the patch sequence is voted by its image patches in the sequence. For instance, given a patch sequence, if more than half of its image patches are classified as the drought condition by the CNN method, then it will be classified as the drought condition. Similarly, we define the image sequence classification for the CNN method.

Table A4.2. The classification results of the 15 data sampling strategies. The data sampling strategy is named as $S_m F_p L_q$, which means that the data sequence is sampled with time interval m from the p^{th} (First) image to the q^{th} (Last) image in the sequence.

Sequence length	**Name**	**Patch sequence classification**			**Image sequence classification**		
		Precision	Recall	F-score	Precision	Recall	F-score
10 images	$S_0F_1L_{10}$	56.7%	54.1%	55.4%	55.3%	53.0%	54.1%
	$S_0F_{11}L_{20}$	57.1%	57.7%	57.5%	56.7%	53.9%	55.3%
	$S_0F_{21}L_{30}$	72.7%	72.1%	72.4%	69.4%	69.9%	69.5%
	$S_1F_1L_{20}$	57.6%	58.0%	57.8%	54.3%	54.7%	54.5%
	$S_1F_6L_{25}$	68.9%	69.0%	68.9%	60.2%	62.1%	61.1%
	$S_1F_{11}L_{30}$	71.0%	70.7%	70.9%	68.2%	68.7%	68.5%
	$S_2F_1L_{30}$	65.0%	64.1%	64.5%	64.2%	62.2%	63.2%
15 images	$S_0F_1L_{15}$	57.1%	54.8%	55.9%	56.0%	54.1%	55.0%
	$S_0F_{11}L_{25}$	68.0%	66.2%	67.1%	65.9%	64.4%	65.1%
	$S_0F_{16}L_{30}$	73.4%	71.6%	72.5%	71.1%	70.3%	70.7%
	$S_1F_1L_{30}$	70.8%	71.4%	71.1%	69.2%	70.6%	69.9%
20 images	$S_0F_1L_{20}$	68.8%	67.0%	67.9%	63.2%	62.8%	63.0%
	$S_0F_6L_{25}$	70.7%	71.0%	70.8%	66.1%	67.3%	66.7%
	$S_0F_{11}L_{30}$	**74.1%**	**75.2%**	**74.6%**	**73.0%**	**71.1%**	**2.0%**
30 images	$S_0F_1L_{30}$	74.5%	76.8%	75.6%	74.6%	72.4%	68.8%

4.2. EXPERIMENTS ON OPTIMAL DATA COLLECTION STRATEGIES (ODCS)

Compared to the *LemnaTecDD Dataset* that only collects plant image data daily during ten days, the *MSTCivil Dataset*, which collects image data daily during 30 days, is more suitable for the experiments of finding the ODCS. Therefore, by using the 15 selected data sampling strategies, the proposed BLSTM model is tested on the *MSTCivil Dataset*. The classification performance of these strategies are presented in Table A4.2.

According to the classification results shown in Table A4.2, there are three main observations:

- Unsurprisingly, the $S_0F_1L_{30}$ strategy that uses all the image data in 30 days outperforms all the other 14 strategies, because the temporal variation in the long time period can provide more detailed information to find the unique pattern of the drought stress condition;

Table A4.3. The proposed BLSTM model is validated on *LemnaTecDD Dataset* by comparing to the BRNN model, the LSTM model and the CNN model.

Method	Patch sequence classification			Image sequence classification		
	Precision	Recall	F-score	Precision	Recall	F-score
BLSTM	**80.3%**	**77.3%**	**78.8%**	**78.7%**	**75.4%**	**77.0%**
BRNN	76.1%	77.4%	76.7%	75.2%	73.7%	74.4%
LSTM	72.1%	73.6%	72.9%	70.2%	71.9%	71.0%
CNN	64.4%	62.1%	63.2%	61.1%	60.5%	60.8%

- The $S_0F_11L_{30}$ strategy, which only uses ⅔ of the entire image dataset, achieves really competitive classification performances;

- According to the F-score of image sequence classification of the $S_0F_1L_{10}$ (54.1%), the $S_0F_1L_{15}$ (55.0%) and the $S_0F_1L_{20}$ (63.0%), the image data from the first 10 or 15 days in the 30 days period have minor contribution to the drought plant stress detection. This observation can be confirmed by the $S_0F_16L_{30}$ strategy with the F-score of image sequence classification as 70.7%, which can also achieve good classification performances without the information from the first 15 days.

By considering both the accuracy and the efficiency, the ODCS could be the $S_0F_11L_{30}$ strategy that uses the continuous image data from the 11^{th} day to the 30^{th} day after planting to achieve 74.1% precision and 75.2% recall in the patch sequence classification evaluation, and 73.0% precision and 71.1% recall in the image sequence classification evaluation. According to the classification performances of the $S_0F_1L_{20}$ strategy, the earliest moment that the proposed method can accurately detect the drought stress is the 20^{th} day after planting.

Table A4.4. The proposed BLSTM model is validated on the *MSTCivil Dataset* by comparing to the BRNN model, the LSTM model and the CNN model.

Method	Patch sequence classification			Image sequence classification		
	Precision	Recall	F-score	Precision	Recall	F-score
BLSTM	**74.1%**	**75.2%**	**74.6%**	**73.0%**	**71.1%**	**72.0%**
BRNN	72.5%	71.0%	71.7%	71.4%	70.8%	71.1%
LSTM	70.7%	67.2%	68.9%	70.9%	70.0%	70.4%
CNN	66.1%	63.1%	64.6%	62.9%	60.3%	61.6%

4.3. VALIDATION OF THE BLSTM MODEL

In this section, to validate the proposed BLSTM model, we compare it with the bidirectional RNN (BRNN) model, the LSTM model, and the CNN model on the two RGB image datasets.

4.3.1. Methods to Be Compared. Instead of using LSTM blocks, the BRNN model uses traditional RNNs to process sequential data in two directions. The architecture of the LSTM model is similar to the proposed BLSTM model, but the main difference between them is that the LSTM model can only process data sequences in one direction. For the CNN method, we use the fine-tuned *VGG-16* model Simonyan and Zisserman (2014) (shown in Figure A3.4). The CNN model takes the individual image patch from the patch sequence as the input to conduct the drought plant stress detection without the temporal variation information.

The training processes of the BRNN model and LSTM model are similar to that of the BLSTM model. The parameters are initialized from a $N(0.0, .05)$ distribution (the forget gate bias of LSTMs are added 1 at initialization), and then they are trained with the RMSprop optimizer using 10^{-3} as the learning rate and 10^{-6} as the decay rate. The LSTM model is trained for 50 epochs with early stopping. However, since the BRNN model will be slower to converge than the LSTM model, we train the BRNN model for 300 epochs and use early stopping based on the validation performance. For the CNN model, we use the fine-tuned *VGG-16* model Simonyan and Zisserman (2014) (shown in Figure A3.4) introduced in the feature extraction section. To test this CNN model, the input of this model is the 224×224 image patch in the patch sequence, where the image patch will inherit the label from the patch sequence to which it belongs.

4.3.2. Comparison of Classification Performances. The quantitative comparison results on the *LemnaTecDD Dataset* are presented in Table A4.3. Based on the selected ODCS ($S_0F_11L_{30}$), we compare the proposed BLSTM model with other models on the *MSTCivil Dataset*, whose results are presented in Table A4.4.

As shown in Table A4.3 and Table A4.4, the proposed BLSTM model outperforms all the other models in both of the patch sequence and image sequence classification evaluations on the two plant image datasets. Since the CNN model only considers the spatial features in the individual image patch without any temporal variation information, it does not work well with the drought samples in the early stage. Compared with the CNN model, the LSTM model is able to use the variation information during the plant growth to identify the drought samples in the early stage. However, due to the subtle variation of the plants with the mild drought condition (some samples shown in Figure A5.1), the LSTM model will make mistakes in the classification of the mild drought samples. Compared with the LSTM model, by using the bidirectional mechanism, the proposed BLSTM model and the BRNN model, which can learn the full context information in the temporal variation pattern, are able to recognize most of the mild drought samples. The BLSTM model is slightly better than the BRNN model in the classification performance. But the BRNN model needs more training time to obtain a good classification performance than the BLSTM model, due to the vanishing gradient problem.

Compared to the competitive classification performances on the *LemnaTecDD Dataset*, the BLSTM model achieves inferior performance (but still competitive compared to other methods) in the *MSTCivil Dataset*. The main reason is the strong interference in the *MSTCivil Dataset* (shown in Figure A5.2), which is collected in a greenhouse that needs to be used for several experiments simultaneously. In some images, some plants are out of view for recording measurements, which cause inconsistency to the image data. Since the image data are collected using the natural light source, shadows and nonuniform illumination conditions add challenging interferences to the early drought plant stress detection task. In addition to these two main problems, some other interferences, such as occlusions and water stain reflections, add nonuniform noises to the image data, which are also challenging problems to the drought plant stress detection task on RGB images.

5. CONCLUSIONS

Early drought plant stress detection is of great relevance in precision plant breeding and production. However, the previous methods based on hyperspectral image analysis were mostly focusing on analyzing the relationship between spectral reflectivity and the leaf water content on individual hyperspectral images, but they ignored the temporal variation information of the plants under the drought stress condition. In addition, the applications of these approaches are limited by the high cost of hyperspectral imaging systems. In this work, we apply the Bidirectional Long Short-Term Memory (BLSTM) networks to RGB image datasets for early drought plant stress detection for the first time. By using LSTM memory blocks and the bidirectional mechanism, the proposed BLSTM model is able to use the discriminative temporal variation information and the full context information in the early plant growth stage for the classification of a patch sequence containing plants under the drought condition or not. Two independently collected RGB image datasets are used for the validation of the proposed method. Optimal data collection strategies in a given environment are also investigated to efficiently detect the drought stress in the early stage.

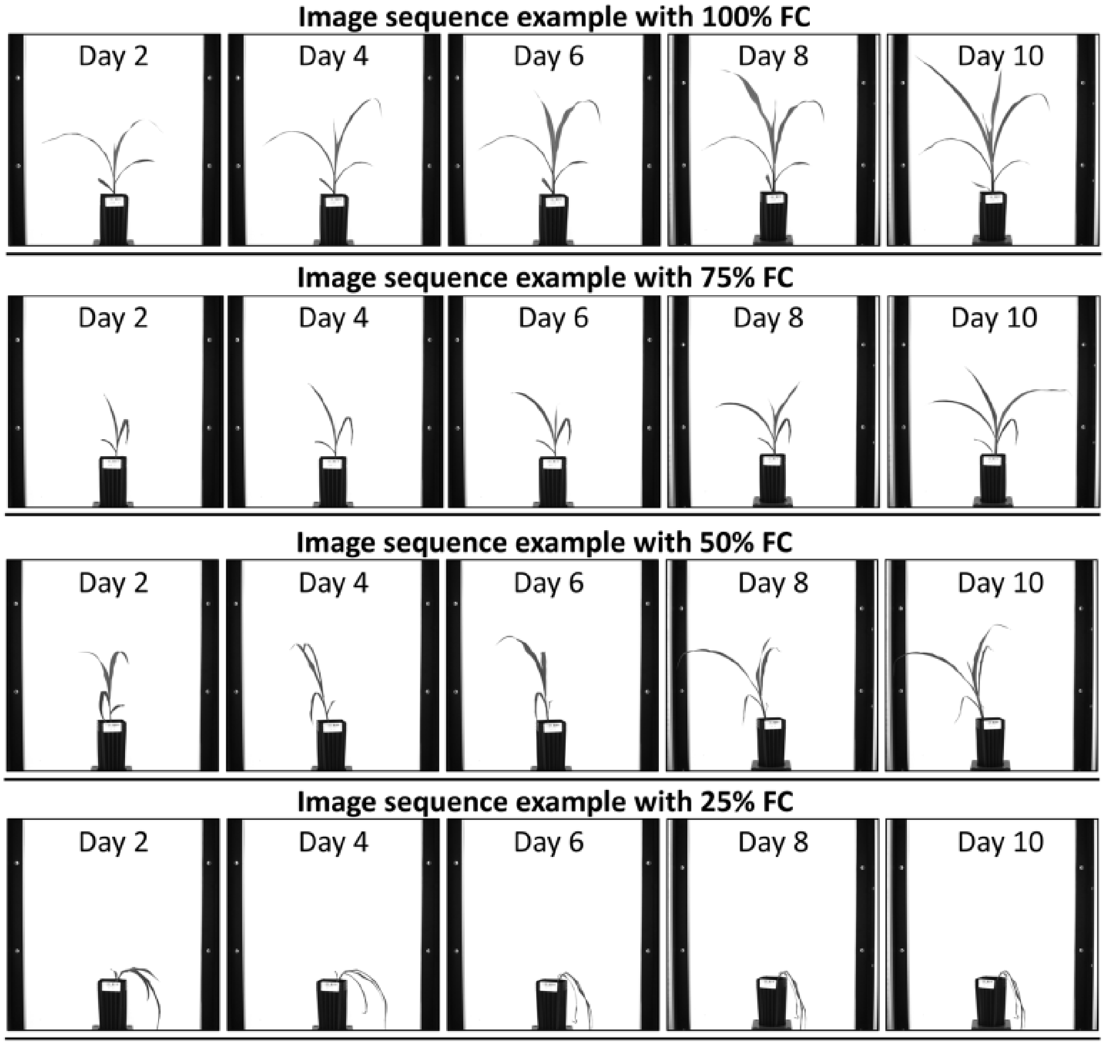


Image sequence example with 100% FC
Day 2
Day 4
Day 6
Day 8
Day 10
Image sequence example with 75% FC
Day 2
Day 4
Day 6
Day 8
Day 10
Image sequence example with 50% FC
Day 2
Day 4
Day 6
Day 8
Day 10
Image sequence example with 25% FC
Day 2
Day 4
Day 6
Day 8
Day 10

Figure A5.1. Image sequence samples of the typical control plant (100% FC) and the typical drought plants (75% FC , 50% FC and 25% FC). The mild drought plant with 75% FC watering regimes, which has very minor drought symptoms, is really challenging for the drought stress detection task.

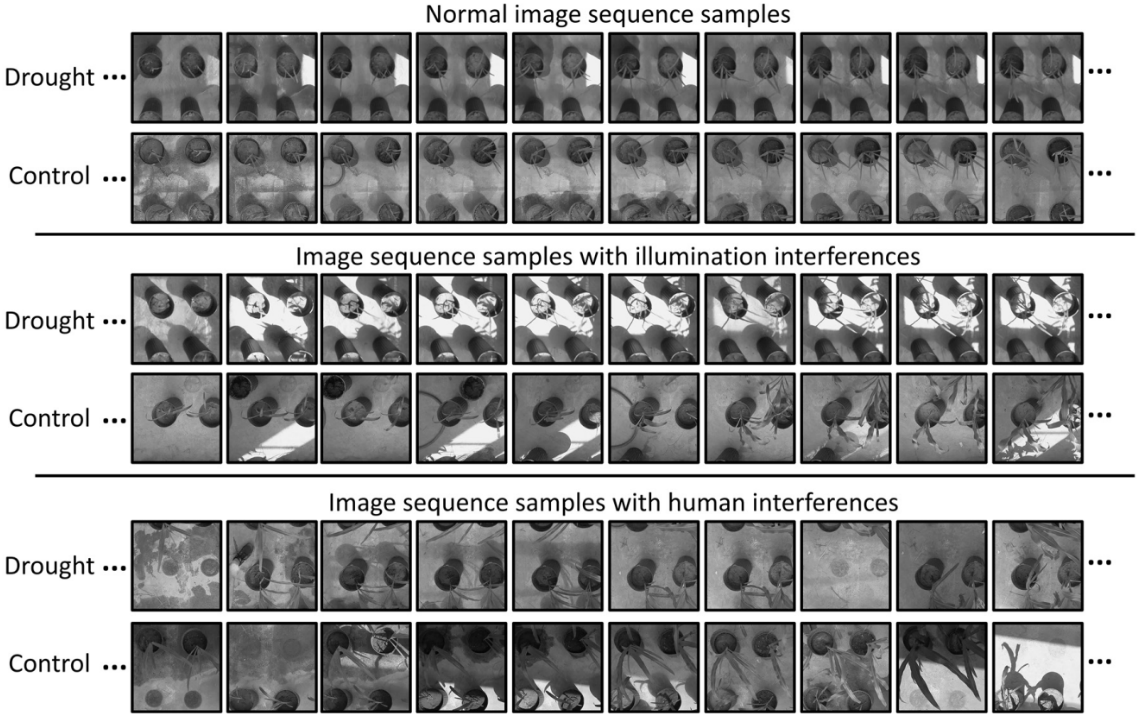

Normal image sequence samples
Drought •••
•••
Control •••
•••
Image sequence samples with illumination interferences
Drought •••
•••
Control •••
•••
Image sequence samples with human interferences
Drought •••
•••
Control •••
•••

Figure A5.2. Image sequence samples in the *MSTCivil Dataset*. Samples with illumination and human interferences are challenging problems to our proposed method.

APPENDIX B.

A HIERARCHICAL CONVOLUTIONAL NEURAL NETWORK FOR VESICLE FUSION EVENT CLASSIFICATION

1. INTRODUCTION

Vesicle exocytosis is an essential cellular trafficking process, by which materials (e.g., transporters, receptors and enzymes) are transported from one membrane-bounded organelle to another or to the plasma membrane for growth and secretion. Vesicle exocytosis needs to be highly regulated since its dysregulation is related to many human diseases (e.g., neurodegenerative disease, cancer and diabetes)Hou and Pessin (2007)Jahn and Fasshauer (2012). Different modes of vesicle exocytosis have been found and characterized in mammalian cells. These include the *full fusion* where a vesicle collapses completely when it fuses with the plasma membrane, and the *partial fusion* or “kiss-and-run” fusion where a vesicle transiently fuses with the plasma membrane without the full collapse Rizzoli and Jahn (2007)Xu *et al.* (2011a). In cell biology research, it is of great importance to detect vesicle fusion events and also to classify different modes of vesicle exocytosis. Because the quantitative analysis of these biological processes can provide insights into cellular behaviors in normal and disease conditions.

Total Internal Reflection Fluorescence Microscopy (TIRFM), which illuminates the aqueous phase immediately adjacent to a glass interface with an exponentially decaying excitation (about 100 nm in z-axis), has been used widely to visualize single vesicle exocytosis at the cell surface Axelrod (1981a)Schneckenburger (2005a). A pH-sensitive mutant of GFP, pHluorin, was developed and expressed to visualize vesicle exocytosis Miesenböck *et al.* (1998). Usually, pHluorin is targeted to the lumen of the vesicle, which is quenched and non-fluorescent in acidic environment, but becomes brightly fluorescent when the vesicle exposes to the extracellular neutral environment as the vesicle fuses with the plasma membrane Xu *et al.* (2011a)Xu *et al.* (2016). In this study, we imaged a variety of vesicle exocytosis in different types of mammalian cells. These include constitutive exocytosis (transferrin receptor-pHluorin exocytosis in endothelial cells and 3T3-L1 adipocytes) and regulated exocytosis (VAMP2-pHluorin labeled insulin granule in MIN-6 cells and VAMP2-pHlurin labeled GLUT4 vesicle in 3T3-L1 adipocytes). Quantitative analysis of the vesicle exocytosis in these typical examples will strengthen our understanding of how vesicle exocytosis is regulated and how its dysregulation triggers human disease (e.g., insulin resistance and diabetes)Bornemann *et al.* (1992a)Leney and Tavare (2009a)Xu *et al.* (2011a).

Usually, the membrane fusion between pHluorin labeled vesicles and the plasma membrane can be represented by 2 significant stages in a continuous video sequence, as illustrated in Figure A1.1 In stage 1, the vesicle is invisible in the *pre-appearance frame* (quenched), and then suddenly appears in the *first-appearance frame* as a brightly fluorescent circle spot. In stage 2, after being immobilized for some frames (from about 100 ms to a few seconds), the vesicle will either fuse completely with the plasma membrane with a visible bright "halo" (full fusion event), or remain its circular shape and gradually fade (partial fusion event), which can be observed in the *last appearance frame*, respectively. At the end of this process, the vesicle under the full or partial fusions will disappear in the *disappearance frame*. Note that, since the moving trajectory of vesicles during the exocytosis process is almost perpendicular to the cell membrane, the trajectory projected onto the cell surface (i.e., the image plane in the TIRFM) only has a small spatial displacement. In this movement process, the appearance variation pattern of the vesicle fusion event is a critical characteristic that is able to generate representative features to distinguish the vesicle fusion event from the background. Specially, the *pre-appearance frame*, *first-appearance frame*, *last-appearance frame* and *disappearance frame* are the 4 key moments of the vesicle fusion event, which represent the significant appearance change of a given fusion event.

A typical time-lapse TIRFM movie consists of thousands of individual frames with hundreds of vesicle fusion events. Unfortunately, so far the vesicle fusion detection and classification are performed mainly in a manual manner, which is a very time-consuming process, and likely to introduce personal biases. Therefore, there is a great demand to

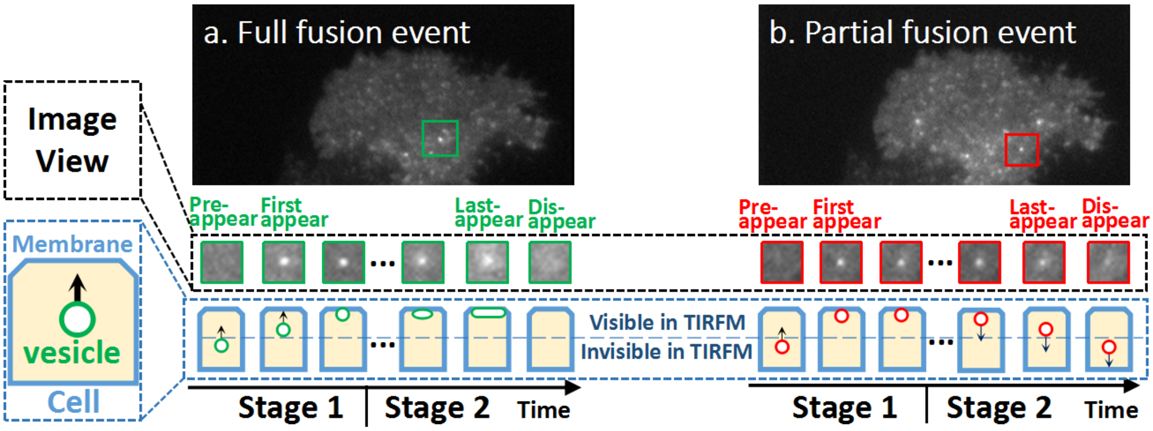


Figure A1.1. The 2 significant stages of vesicle fusion processes and the related 4 key moments. 3T3-L1 adipocytes were transfected with VAMP2-pHluorin to label the GLUT4 vesicles. pHluorin is a pH-sensitive fluorescent protein that is invisible in the lumen of acidic vesicles, which becomes much more fluorescent when a vesicle fuses with the plasma membrane and exposes to a neutral environment. After a vesicle touches the cell membrane, it either fully collapses and fuses with the plasma membrane (a. Full fusion event), or partially fuses with the plasma membrane and then is retrieved rapidly by the clathrin-dependent process (b. Partial fusion event).

develop effective computational tools to automatically extract the vesicle fusion event information in TIRFM video sequences, which will aid the quantitative analysis on the vesicle exocytosis process.

1.1. RELATED WORK

When the computer-based microscopy image analysis is used to relieve human from the tedious manual labeling Basset *et al.* (2014a)Basset *et al.* (2015a)Godinez *et al.* (2009), it is unsurprising that lots of challenges, such as the uncontrollable noise interference of TIRFM images and the high variability of fusion events' properties (e.g., intensity profiles, lifetime length and movement patterns), hinder the automated image processing. Furthermore, some of the bright spots (endocytic vesicles or vesicles from other non-acidic compartments) in TIRFM image sequences are moving in and out of the TIRFM field,

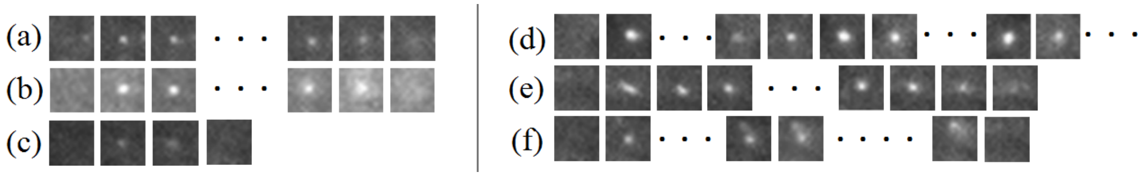


Figure A1.2. Some samples of partial fusion event (a), full fusion events (b,c), and non-fusion events (d,e,f). (a) A typical partial fusion event; (b) A typical full fusion event with the “puff”phenomenon; (c) A short full fusion event is characterized by its “puff”phenomenon; (d) A bright circular object caused by the background intensity fluctuation; (e) A moving bright spot, which only moves in the first several frames then stays immobile, is similar to a partial fusion event when it stops moving; (f) A background fluctuation, which is really similar to standard full fusion events in the early stage, then gradually moves out of the field of view.

which is a great challenge for designing automated algorithms for vesicle fusion detection. In order to detect fusion events, one needs to use specific detection algorithms considering both spatial and temporal features of individual objects.

Based on the bright circular appearance of vesicle fusions under the TIRFM, some approaches have been proposed to perform automated fusion identification, such as the pixel intensity thresholding methods in Huang *et al.* (2007a)Yuan *et al.* (2015) and the intensity distribution analysis methods in Smith *et al.* (2011)Wu *et al.* (2015a). However, these methods are sensitive to the variation of vesicle fusion intensity profiles (shown in Figure A1.2(a,b,c)). In order to improve the tolerance to the variation, some automated approaches were developed to model the moving process of fusion events. Based on both the temporal and spatial features, a template matching method was proposed to identify the fusion events with high correlation to a standard fusion event template in Vallotton *et al.* (2007a). In another study, a Gaussian model was used to fit typical fusion events in Bai *et al.* (2007a), where the parameters in the Gaussian model are used to classify fusion events. However, due to the frequent background intensity fluctuations (as shown in Figure A1.2(d, e, f)) introduced by the TIRFM system and intracellular activities, it is hard to build a standard template or a general model to represent all fusion events.

Because of the large variations of the fusion events' properties (e.g., intensity profiles, lifetime length and movement patterns) and frequent background fluctuations, the robustness of a vesicle fusion detection and classification method is highly important. A robust detection method was proposed in Berger *et al.* (2012), which first detects candidate fusion events that suddenly appear in the TIRFM field. Then, a diffusive model is developed to analyze the intensity distribution variation pattern of the fusion event for the classification. Based on the visible "puff"phenomenon of the full fusion event, the diffusive fusion model effectively distinguishes full fusion events from non fusion regions, leaving a large amount of partial fusion events unrecognized. In addition, a Layered Probabilistic Approach was proposed in Godinez *et al.* (2012) to identify full fusion events by exploring three abstractions: the intensity over time, the underlying temporal intensity model and the high level behavior. Each of these three abstractions corresponds to a layer and these layers are represented via stochastic hybrid systems and hidden Markov models. However, partial fusion events are not considered in this work.

Unlike the full fusion event, which can be distinguished by its "puff"/spread signal, the partial fusion event is resembled to other bright spots (Figure A1.2(d, e, f)) on the background, which is problematic in most of the existing detection and classification methods. In order to reveal the unique variation pattern of the fusion events, a learning based method was developed in our previous work Li (2015a). An adaptive detection and tracking method is first applied to TIRFM images to search for potential fusion patches through video frames, then a Gaussian Mixture Model (GMM) is fitted on each individual fusion event. Using the estimated parameters of this model as features, a classifier is trained to distinguish full fusion events, partial fusion events and non-fusion events. However, in this GMM-based method, the handcrafted features ignore the discriminative appearance information from the 4 key moments of a fusion event, which leads to miss-detection problems in short fusion events (shown in Figure A1.2).

1.2. THE MAJOR CHALLENGES

According to the observation of our own datasets and the review of previous works, the major challenges to the task of detecting and classifying vesicle fusion events are summarized as follows.

1.2.1. The High Variability of Vesicle Fusion Events. Some typical partial fusion events and full fusion events are shown in Figure A1.2(a) and Figure A1.2(b) respectively, from which we can observe the characteristics of vesicle fusion events. For example, normally partial fusion events present the momentary appearance and disappearance, and full fusion events present a sudden appearance and a gradual disappearance with their signals fading away. However, in practical cases, the vesicle fusion event has large variations in its intensity profile, lifetime and movement pattern. For instance, compared with a typical full fusion event in Figure A1.2(b), the full fusion event in Figure A1.2(c) has a much shorter lifetime and a much more blurry intensity profile. These variations yield challenges in modeling the various visual patterns of fusion events.

1.2.2. Complex Background Interferences. Besides vesicle fusion events, there exist a large amount of other bright circular spots on the background, which are challenges for automated fusion event detection and classification. For instance, the circular background intensity fluctuation (Figure A1.2(d)) is similar to a partial fusion event. Some moving bright spots, which are temporarily immobile near the cell membrane for several frames (Figure A1.2(e,f)), can be mistakenly classified as partial fusion events. These interferences yield challenges in selecting effective features to build discriminative classifiers.

1.3. OUR PROPOSAL AND CONTRIBUTIONS

Rather than designing handcrafted visual models or features, Convolutional Neural Networks (CNN) that can learn the discriminative features from big training data have been widely used in different real world classification tasks, such as image recognition Krizhevsky *et al.* (2012a)Lawrence *et al.* (1997), video analysis Yue-Hei Ng *et al.* (2015)Karpathy *et al.* (2014) and natural language processing Hu *et al.* (2014)Kim (2014). CNN is a promising learning based method to handle classification challenges on microscopy images, such as cell detection Mao and Yin (2016)Mao *et al.* (2016). Therefore, in order to enhance the tolerance to the variation of fusion events and the unpredictable background interferences, we propose to develop a novel CNN-based application which applies a Hierarchical Convolutional Neural Network (HCNN) to explore both appearance features and temporal cues for the vesicle fusion event classification. First, we extract fusion event candidate sequences and their appearance features from the input video data by using a newly developed iterative tracking algorithm. Secondly, a center-surrounded Gaussian Mixture Model (GMM) is fit on each patch of the patch sequence using the RANSAC algorithm Fischler and Bolles (1981) to remove outliers during the fitting process. The patch sequences are aligned with the same time length and time-series intensity change features corresponding to the Gaussian models' parameters are extracted over time. Thirdly, based on the time-series parameters from Gaussian Mixture Models and 4 key moments of the fusion event candidate sequence, a HCNN is developed to automatically select discriminative temporal and appearance features for the classification of the fusion event candidates in challenging datasets with low Signal-to-Noise-Ratio and frequent background fluctuations.

Our contributions in this paper include: (1) A novel application is proposed to detect and classify vesicle fusion events. The Hierarchical Convolutional Neural Network (HCNN) is utilized to learn discriminative appearance features from 4 key moments of a fusion event and combine them with the temporal features from the parametric Gaussian Mixture Models over time; (2) A center-surrounded Gaussian Mixture Model is used to model the intensity profile change of a fusion event in its entire lifetime; (3) A newly developed vesicle fusion event tracking algorithm is applied for the appearance feature extraction.

The rest of this paper is organized as follows: in Section 2, we briefly introduce our newly developed vesicle fusion event tracking algorithm, which contributes to appearance feature extraction for fusion event classification; in Section 3, the classification of the fusion event candidates by HCNN is presented; in Section 4, we validate our method on 9 challenging datasets and compare it with the previous methods and other neural network architectures. The paper concludes with Section 5.

2. DETECTION AND TRACKING ALGORITHM

Based on our preliminary work on detecting and tracking vesicle candidates in video sequences Li (2015a), we improved the tracking algorithm to accurately measure the lifetimes of vesicle fusion events, which is important for the feature extraction task in fusion event classification. The major goal of our new tracking algorithm is to find the *first-appearance frame* and the *last-appearance frame* of a potential fusion event and every patch center between the *first-appearance frame* and the *last-appearance frame*. We utilize Figure A2.1 to illustrate how to iteratively search in the forward direction to find the *last-appearance frame* (the search in the backward direction to find the *first-appearance frame* is similar).

Assume we find the pixel (x^*, y^*) with the local maximum of local contrast as the center of the potential fusion event and crop an $n \times n$ image patch around it. Since we use fixed size patches, we only need to record the coordinates of the patch center in the fusion event candidate patch sequence, which are denoted as $S = \left\{x_t^*, y_t^* | t \in [t_{first}, t_{last}]\right\}$ where t_{first} and t_{last} denote the first and last frame index of the patch sequence, respectively. At the beginning, $t_{first} = t_{last} = t_0$. During each iteration, we search the *last-appearance frame* in a sliding temporal window of D frames. Three situations are considered during the iterative search:

Situation 1, if the maximums of the local contrast in all D frames around location $(x^*_{t_{last}}, y^*_{t_{last}})$ are larger than ε, so we can update $S = \left\{x_t^*, y_t^* | t \in [t_{first}, t_{last}]\right\}$ by setting $t_{last} \leftarrow t_{last} + D$. Then, we continue the search from frame $t_{last} + 1$ to frame $t_{last} + D$.

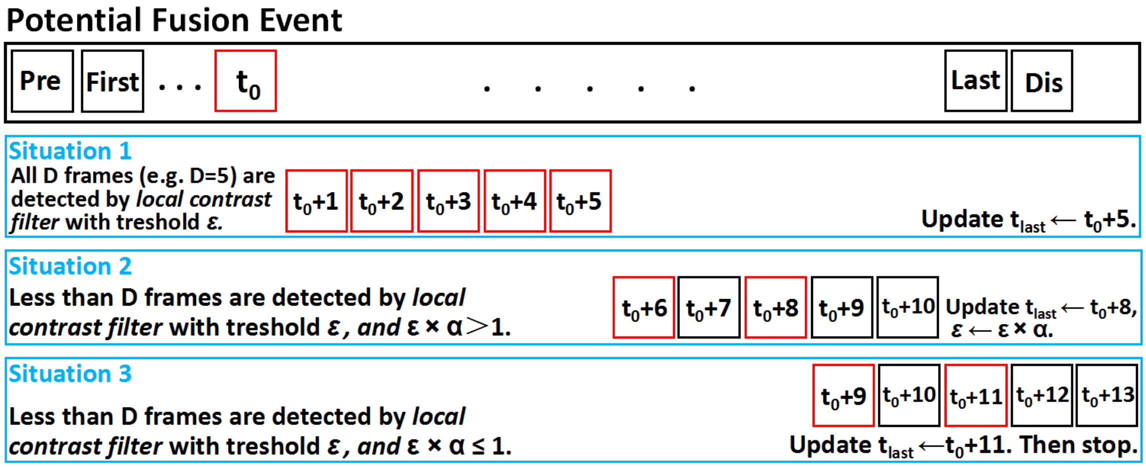

Potential Fusion Event
Pre
First
. . .
t_0
. . . . . .
Last
Dis
Situation 1
All D frames (e.g. D=5) are detected by *local contrast filter* with treshold ε.
t_0+1
t_0+2
t_0+3
t_0+4
t_0+5
Update $t_{last} \leftarrow t_0+5$.
Situation 2
Less than D frames are detected by *local contrast filter* with treshold ε*, and* ε × α>1.
t_0+6
t_0+7
t_0+8
t_0+9
t_0+10
Update $t_{last} \leftarrow t_0+8$, $\varepsilon \leftarrow \varepsilon \times \alpha$.
Situation 3
Less than D frames are detected by *local contrast filter* with treshold ε*, and* ε × α ≤ 1.
t_0+9
t_0+10
t_0+11
t_0+12
t_0+13
Update $t_{last} \leftarrow t_0+11$. Then stop.

Figure A2.1. An example to search the candidate patch sequence S in the forward temporal direction.

Situation 2, if not all of the maximums of the local contrast in D frames around location $(x^*_{t_{last}}, y^*_{t_{last}})$ are larger than ε, while $\varepsilon \times \alpha \ > 1$ (α is a decay rate on the threshold), we update t_{last} as the last frame within the D frame whose maximal local contrast is larger than ε and the patch centers are updated accordingly. The threshold is updated as $\varepsilon \leftarrow \varepsilon \times \alpha$. Then, we continue the search from frame $t_{last} + 1$ to frame $t_{last} + D$.

Situation 3, if not all of the maximums of the local contrast in D frames around location $(x^*_{t_{last}}, y^*_{t_{last}})$ are larger than ε and $\varepsilon \times \alpha \ \leq 1$, we update the patch sequence similar to situation 2, then we stop the iteration.

By applying this iterative tracking algorithm to the TIRFM image sequence, we can obtain the whole lifetimes of potential fusion events in the format of candidate patch sequences, each of which records the coordinates of the patch center from the *first-appearance frame* to the *last-appearance frame*. For each potential fusion event, we compute the pairwise Euclidean distance between each consecutive pair of patch centers within the candidate patch sequence. If any of these distances is larger than the neighborhood size n, this candidate patch sequence is highly possible to be a non-fusion event caused by a moving object from the background, and we remove it from the candidate list.

3. CLASSIFICATION OF FUSION EVENT CANDIDATES

In this section, we will introduce the classification of fusion event candidates by using a novel Hierarchical Convolutional Neural Network (HCNN). Compared with the Support Vector Machine-based classification method in Li (2015a), HCNN is able to automatically select discriminative features which can provide the comprehensive representation of the fusion event. In order to enhance the tolerance to the variation of fusion events and the unpredictable background interferences, the proposed HCNN architecture considers both spatial and temporal information. The input of our HCNN consists of the time-series parametric information from the Gaussian Mixture Model fitting, and the visual appearance information from the 4 key moments of the fusion event candidate. The former is aiming at revealing the unique hidden variation pattern of the vesicle fusion event in its entire lifetime. The latter is proposed to extract the extraordinary visual appearance features of the vesicle fusion event. Moreover, the hierarchical architecture is able to exploit the high-level abstraction of intensity profiles of individual frames and the high-level temporal features from the entire fusion event lifetime to accurately distinguish fusion events from the other similar circular bright spots in Figure A1.2.

3.1. DATA PREPARATION

Because of the frequent background interferences in the TIRFM video data, directly thresholding the candidate patch sequence might not be a good option to present its intensity profile variation. Therefore, we adopt the data preparation strategy in our previous work Li (2015a). First, a robust Gaussian Mixture Model (GMM), which consists of two center-surrounded 2D Gaussian models ($\mathbf{Area}_p$ and $\mathbf{Area}_f$ in Figure A3.1), is adopted to fit the intensity profile of each fusion event candidate, where a Random Sample Consensus algorithm Fischler and Bolles (1981) is applied to robustly estimate the parameters of

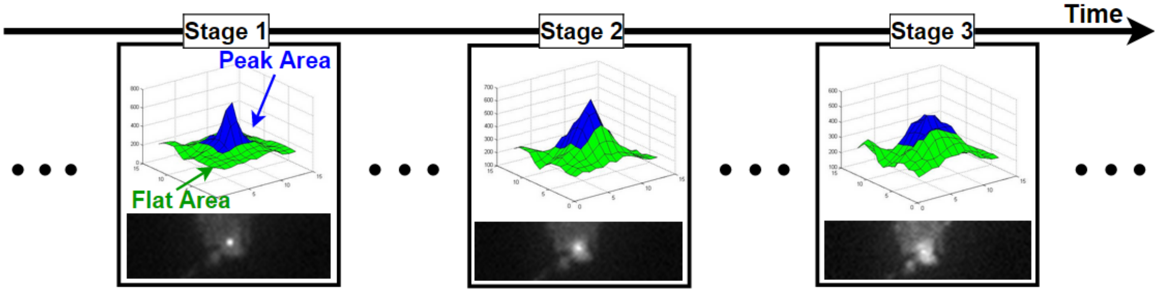


Figure A3.1. The Gaussian Mixture Model consists of a 5 × 5 "peak area" and a 13 × 13 "flat area".

Gaussian models without the outlier effect. Second, since most of the fusion events have their lifetimes less than 24 frames in the datasets we used in this study, we extract 24 image patches from each fusion event candidate starting from the *first-appearance frame*. For those fusion event candidates whose lifetimes are shorter than 24 frames, we will zero-padding them. For those fusion event candidates with longer lifetimes, they will be cut into the time length. Third, for each fusion event candidate, there are 24 extracted image patches in the patch sequence, where each image patch is represented by a set of GMM parameters (λ_{peak}[1], μ_{peak}, σ_{peak} of $\mathbf{Area}_p$, and $\lambda_{flat}, \mu_{flat}, \sigma_{flat}$ of $\mathbf{Area}_f$). Thus, the time-series intensity profile change of a vesicle fusion event candidate, which is represented by 24 sets of GMM parameters, can be utilized for fusion event classification.

3.2. THE VARIATION PATTERN IN GMM IMAGE

In order to explore the hidden correlations among the image patches in each fusion event candidate, we generalized the vectorization process in our previous work Li (2015a) by transforming the parameter sets of a fusion event candidate into a 2D image, which concatenates the time-series parameter sets into a 2D array in a special order, as shown in Figure A3.2. We call this 2D array of Gaussian Mixture Model fitting parameters as *GMM image* that allows the HCNN to discover the hidden correlation among the parameter sets.

[1] λ is the weighting coefficient of each Gaussian component in the GMM.

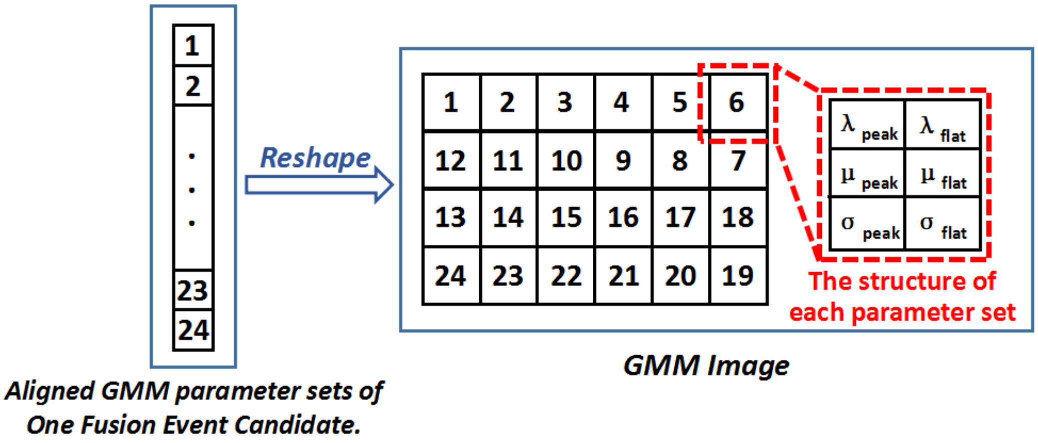


Figure A3.2. Transforming the time-series Gaussian fitting parameter sets to a 2D array (Gaussian Mixture Model image, *GMM image*). In the *GMM image*, each cell represents a parameter set for one image patch of the fusion event candidate. In each cell, the 6 parameters are organized as a 3 × 2 matrix ($\lambda_{peak}, \lambda_{flat}; \mu_{peak}, \mu_{flat}; \sigma_{peak}, \sigma_{flat}$). So the *GMM image*, which contains 24 cells, is a 12 × 12 matrix.

Furthermore, in Figure A3.2, we design the *GMM image* to be a square image, so each parameter set has more chances to be neighboring to other parameter sets. For example, given 24 parameter sets to stitch, if they are concatenated into a 24 × 1 matrix pattern, there is no 4- or 8-connected neighborhood relationship among the parameter sets. However, if we stitch them into a 12 × 2 matrix pattern, the relationship among the parameter sets will increase a little. Thus, in this work, we concatenate the 24 parameter sets into a 4 × 6 matrix pattern, many 4- or 8-connected neighborhood relationships can be built among the parameter sets.

3.3. THE VISUAL APPEARANCE IN 4 KEY MOMENTS

In addition to the *GMM image*, which contains the high-level abstraction of intensity profiles of individual frames, we also consider the appearance features in the 4 key moments of a fusion event candidate. As described in Figure A1.1, the movement of vesicles can be well represented in the 4 key moments: *pre-appearance frame*, *first-appearance frame*, *last-appearance frame* and *disappearance frame*. By using our newly developed vesicle fusion event tracking method, the whole entire of each fusion event candidate is able to be obtained. Therefore, for each candidate, we extract image patches in these 4 key moments. The *first-appearance frame* patch and *last-appearance frame* patch are extracted from the first frame and the last frame in the fusion event lifetime, respectively. The *pre-appearance frame* patch is extracted from the previous frame of the *first-appearance frame*. The *disappearance frame* patch is extracted from the next frame of the *last-appearance frame*. Both the parametric information from the *GMM image* and the 4 image patches of the 4 key moments will be input to the HCNN.

3.4. THE ARCHITECTURES OF OUR HCNN

The overall architecture of our Hierarchical Convolutional Neural Network (HCNN) is shown in Figure A3.3. In the first layer, the inputs of the first 4 Convolutional Neural Networks CNN_1^j ($j \in [1,4]$) are the cropped image patches from 4 key moments, which provide the detailed visual appearance information of fusion event candidates. Each of these four CNNs takes a single cropped image patch. The input of the CNN_1^5 is the *GMM image* which provides the time-series intensity change information of the fusion process (a high-level abstraction using the parameters from Gaussian Mixture Model fitting). In the second layer of our HCNN, we design the CNN_2^6 to learn joint features of the CNN_1^j ($j \in [1,4]$), which indicate the correlation of fusion event patches in the 4 key moments. In the third layer, the combined appearance and time-series intensity change features are fed into the CNN_3^7 to make the final prediction. In our notation of CNN_i^j, i denotes the layer in our HCNN and j indexes the CNN out of the total 7 CNNs in our proposed HCNN architecture.

The design of our proposed HCNN architecture has three motivations. First, the intensity variation pattern of a fusion event, which is different from other bright circular spots in TIRFM image sequences, is a significant characteristic to classify fusion events.

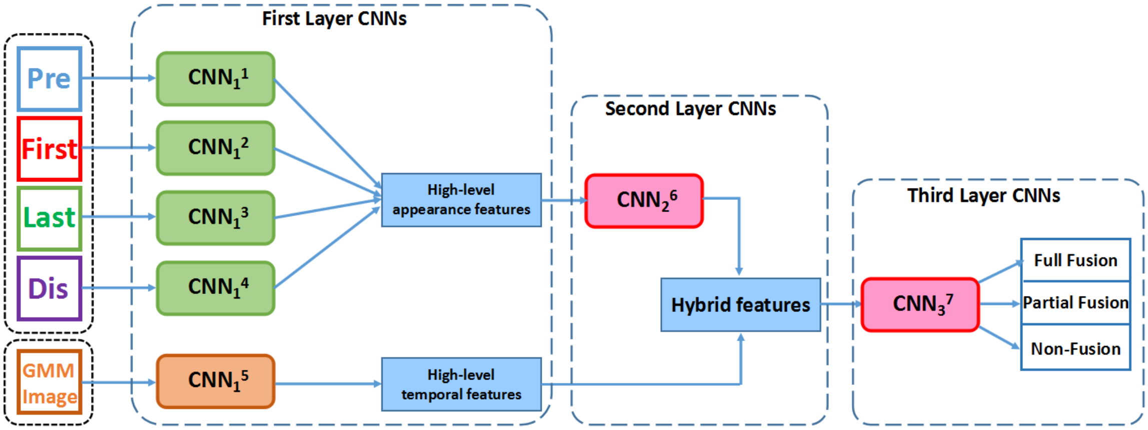


Figure A3.3. The overall architecture of our proposed Hierarchical Convolutional Neural Network (HCNN).

Instead of directly using the consecutive image patch sequence to provide this time-series intensity change information, the time-series parameter sets from Gaussian Mixture Model fitting, which can avoid outlier pixels with undesired intensity fluctuations, are more reliable and the proposed *GMM image* can further explore hidden relations among the time-series parameters. Second, the characteristics of a fusion event's appearances can be well represented in the 4 key moments, thus utilizing these appearance characteristics and the correlation among the 4 key moments should boost the classification performance. Third, our proposed HCNN architecture is able to learn the correlation among the 4 key moments before combing the appearance and temporal features, which can reveal the unique variation pattern of the fusion event.

The first layer of our HCNN contains 5 CNNs ($CNN_1^j, j \in [1,5]$). The first 4 CNNs ($CNN_1^j, j \in [1,4]$), each of which takes a cropped image patch (13×13) of the fusion event in one of the 4 key moments as the input, share the same architecture as shown in Figure A3.4. In the architecture of CNN_1^j ($j \in [1,4]$), there are two Convolutional Layers where each of them is connected to a Rectified Linear Unit (ReLU) for sparse representations. The first Convolutional Layer is followed by a 2×2 Max Pooling Layer with stride 2. The

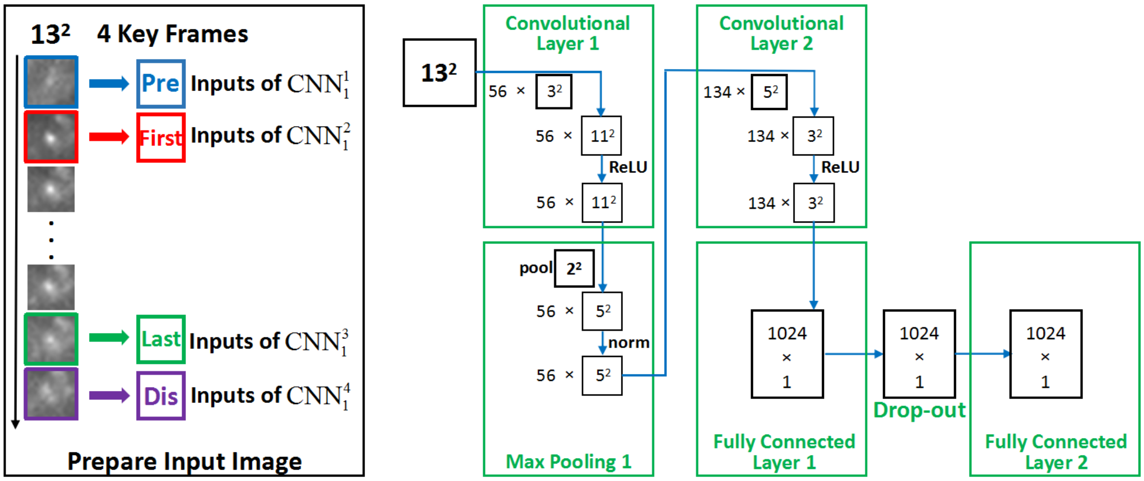


Figure A3.4. The architecture of CNNs (CNN_1^j, $j \in [1,4]$) in the first layer. The inputs of this architecture are image patches which are centered at the maximum intensity pixels of the fusion event in the 4 key moments respectively. In the Convolutional Layer 1, we set the number of the 3×3 kernels as 56. In the Convolutional Layer 2, we set the number of the 5×5 kernels as 134. In the Max Pooling 1, there is a 2×2 max pooling layer with stride 2. The number of neurons in each Fully Connected Layer is 1024.

major goal of adding Max Pooling Layer is to enhance the robustness of the classifier by bringing invariance to the training process. We add a Drop-out Layer Srivastava *et al.* (2014) between the two Fully Connected Layers to avoid the over-fitting.

The CNN_1^5, whose architecture is shown in Figure A3.5, learns the high-level time-series features from the intensity variation pattern introduced by the *GMM image*. There are 3 Convolutional Layers, where each Convolutional Layer is followed by a Rectified Linear Unit (ReLU) for sparse representations. Compared with the other 4 CNNs in the first layer, there is no Max Pooling Layer in CNN_1^5. Because we do not expect to loss any time-series variation information during the convolution. To avoid the over-fitting, we add one Drop-out Layer between the Fully Connected Layer 1 and Fully Connected Layer 2.

The architecture of the CNNs in the second and last layer of our HCNN (CNN_2^6 and CNN_3^7) is shown in Figure A3.6. The input feature layer to CNN_2^6 is the combined feature from the Fully Connected Layer 2 of CNN_1^j ($j \in [1,4]$). The design of CNN_2^6 is to

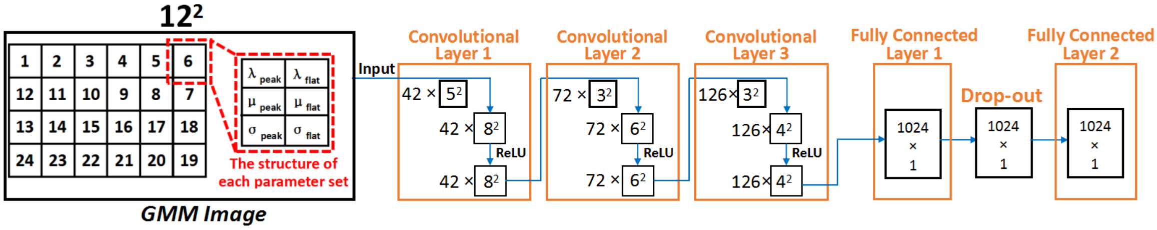


Figure A3.5. The architecture of CNN_1^5 in the first layer of our HCNN. The input of this architecture is the *GMM image*. In the Convolutional Layer 1, we set the number of the 5×5 kernels as 42. In the Convolutional Layer 2, we set the number of the 3×3 kernels as 72. In the Convolutional Layer 3, we set the number of the 3×3 kernels as 126. The number of neurons in each Fully Connected Layer is 1024.

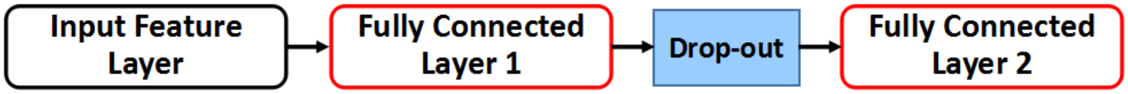


Figure A3.6. The architecture shared by CNN_2^6 in the second layer and CNN_3^7 in the third layer. In CNN_2^6, the input feature layer contains the high-level appearance feature, which is extracted from the 4 key moments. In CNN_3^7, the input feature layer consists of visual and temporal information.

study the correlation information among the 4 key moments before combining appearance features and time-series variation features. The input features to CNN_3^7 is the combined features of the time-series intensity variation features from the Fully Connected Layer 2 of CNN_1^5, and the visual appearance features from the Fully Connected Layer 2 of CNN_2^6. Between the Fully Connected Layer 1 and Fully Connected Layer 2, we add a Drop-out Layer to avoid the over-fitting.

4. EXPERIMENTS

In this section, first we describe our datasets, experimental design and evaluation metrics. Then, we validate the effectiveness of our fusion event candidate extraction. Thirdly, we compare our method with the state-of-the-arts and our previous methods in Li (2015a). Finally, we validate our HCNN design by comparing it with 11 alternative neural network designs.

4.1. DATASETS, EXPERIMENTAL DESIGN AND EVALUATION METRIC.

In this section, we introduce the datasets, experimental design and the evaluation metrics in our experiments.

4.1.1. Datasets. In the experiments, 9 TIRFM image sequences were captured at 5 frame per second (fps), which consist of 15718 frames and 1260 fusion events in total. The detailed information of our datasets is summarized in Table A4.1. All image sequences were well annotated by experienced cell biologists working in the field of vesicle trafficking analysis using TIRFM.

4.1.2. Experimental Design & Evaluation Metric. The leave-one-out strategy is adopted to evaluate the performance of our method, i.e., eight sequences are used for training while the last one is used for testing (the parameters in the detection & tracking process and the Gaussian Mixture Model (GMM) fitting are optimized by the 4-fold cross-validation using the eight training sets). There are totally 9 leave-one-out experiments are performed on the datasets. The average performance on the 9 experiments in terms of precision, recall and F-score is utilized as the evaluation metrics.

Table A4.1. The image size and the number of fusion events in each dataset.

DataSet	**1**	**2**	**3**	**4**	**5**
Full Fusion Event	118	169	31	132	48
Partial Fusion Event	28	64	56	6	10
Image Size (pixels)	327 × 179	271 × 284	233 × 324	271 × 341	408 × 381
DataSet	**6**	**7**	**8**	**9**	
Full Fusion Event	16	19	76	193	
Partial Fusion Event	16	76	11	797	
Image Size (pixels)	382 × 338	241 × 211	478 × 412	485 × 299	

4.2. EFFECTIVENESS OF THE FUSION EVENT CANDIDATE EXTRACTION

By using our newly developed detection & tracking method, we obtain 4642 candidate patch sequences on the 9 datasets. The candidate pool contains all the 1260 ground-truth vesicle fusion events from 15718 frames (i.e., our candidate sequence extraction achieves 100% recall and 27% precision). Instead of exhaustively selecting fusion event candidates from every volume of the TIRFM video sequences, the proposed detection & tracking method not only ensures all vesicle fusion events are included in the fusion event candidate pool, but also effectively improves the efficiency of the whole system. Note, data augmentation techniques (e.g., flipping, rotation and translation) were applied on our positive training samples to provide enough training data.

4.3. COMPARISON WITH THE PREVIOUS METHODS

Our algorithm is compared with the learning-based Gaussian Mixture Model using Support Vector Machine classifier (GMM-SVM, Li (2015a)), the intensity-based Single Gaussian Model (SGM, Bai *et al.* (2007a)) and the Layered Probabilistic Approach (LPA-FullFusion, Godinez *et al.* (2012)). Note, the Layered Probabilistic Approach can not detect partial fusion events. All the parameters in Bai *et al.* (2007a), Godinez *et al.* (2012) and Li (2015a) are optimized to ensure that they can achieve their best performance in our TIRFM

Table A4.2. Comparing our method with 2 state-of-the-arts and our previous method on 9 challenging datasets: GMM-SVM Li (2015a): Gaussian Mixture Model using Support Vector Machine classifier; SGM Bai *et al.* (2007a): Single Gaussian Model; LPA-FullFusion Godinez *et al.* (2012): Layered Probabilistic Approach for full fusion detection.

	Full Fusion			Partial Fusion		
	Precision	Recall	F Score	Precision	Recall	F Score
Our Method	**95.2%**	**96.2%**	**95.7%**	**96.1%**	**96.7%**	**96.4%**
GMM-SVM Li (2015a)	76.9%	79.3%	78.1%	75.5%	76.0%	75.7%
SGM Bai *et al.* (2007a)	61.1%	64.7%	62.8%	64.6%	62.0%	63.3%
LPA-FullFusion Godinez *et al.* (2012)	75.3%	72.3%	73.8%	N/A	N/A	N/A

image sequences. As shown in Table A4.2, compared with the GMM-SVM Li (2015a) that uses handcrafted features, our method achieved much better classification results for both the full fusion event and the partial fusion event in 9 datasets. It validates that the automatically selected features from the time-series intensity change introduced by GMM images and the visual appearance in 4 key moments by our HCNN architecture have a more comprehensive representation of the vesicle fusion event. Compared to the SGM Bai *et al.* (2007a) that only depends on the spatial radius of the Single Gaussian fit to the bright blob, our method achieved better classification results, which proves that the proposed Gaussian Mixture Model has a more precise representation to extract the intensity variation pattern of vesicle fusion events. Compared with the Layered Probabilistic Approach in Godinez *et al.* (2012), which uses three abstractions of fusion events as the feature for the classification, the temporal and spatial features extracted by GMM in our previous method Li (2015a) achieved better classification results on full fusion events. Furthermore, our proposed HCNN architecture, which can automatically select the discriminative features from the whole lifetime of the fusion event, obtained the best performance. In short, besides visual appearance features, our HCNN based method can extract hidden variation patterns of the fusion event, which are qualified for the task of accurate fusion event classification. Fusion event classification samples of our proposed method are presented in Figure A4.6 and Figure A4.7.

4.4. COMPARISON OF DIFFERENT NEURAL NETWORK DESIGNS

In this subsection, first we test different layouts in our overall architecture (Figure A3.3) and compare the performance. Second, we test different input formats of the visual appearance features extracted from 4 key moments and compare the performance. Third, we test different input formats of the temporal features and compare the performance. Last, we test different designs in our individual CNNs (there are 7 CNNs in total, Figure A3.3).

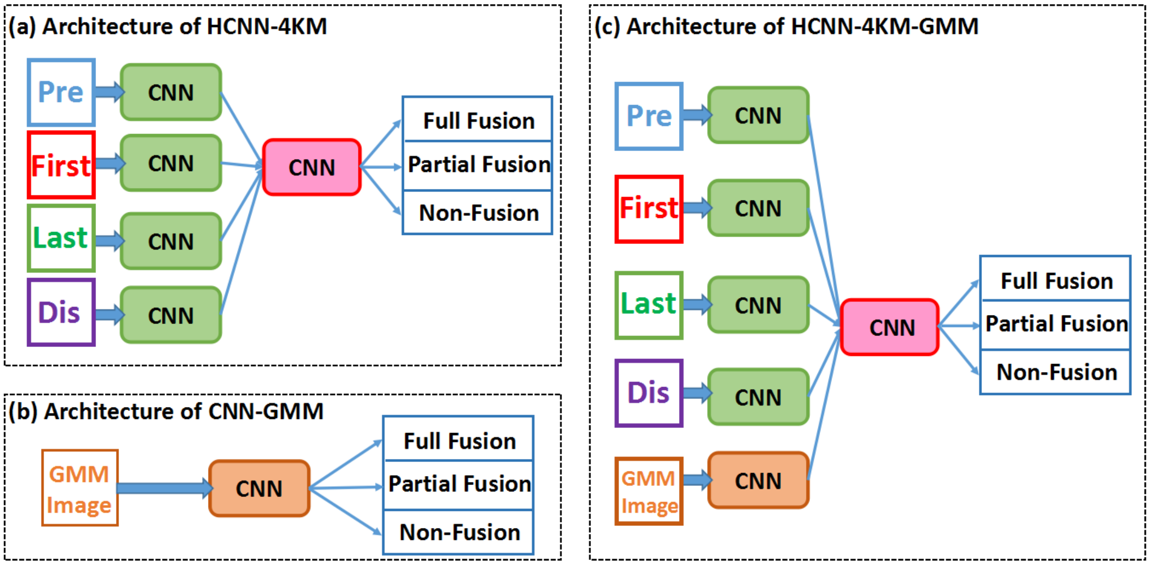


Figure A4.1. The architectures of the HCNN-4KM (a), CNN-GMM (b) and HCNN-4KM-GMM (c).

4.4.1. Comparison of Alternative Overall Architecture Designs.

We designed the HCNN-4KM (Figure A4.1(a)) that only considers appearance features, and the CNN-GMM (Figure A4.1(b)) that only considers temporal features. As shown in Table A4.3, our HCNN architecture outperformed HCNN-4KM and CNN-GMM, which validates that both appearance features and temporal features contribute significantly to the fusion event classification task.

Table A4.3. Comparing our HCNN with 3 alternative overall architecture designs on 9 challenging datasets, which include: HCNN-4KM (Figure A4.1(a)): based on our HCNN, we remove the CNN_1^5 and CNN_2^6 so only appearance features are used; CNN-GMM (Figure A4.1(b)): based on our HCNN, we only use the temporal features in *GMM images* for the classification; HCNN-4KM-GMM (Figure A4.1(c)): based on our HCNN, we remove the CNN_2^6.

	Full Fusion			Partial Fusion		
	Precision	Recall	F Score	Precision	Recall	F Score
Our Method	**95.2%**	**96.2%**	**95.7%**	**96.1%**	**96.7%**	**96.4%**
HCNN-4KM	79.3%	82.4%	80.8%	85.1%	84.7%	84.9%
CNN-GMM	84.8%	88.7%	86.7%	82.0%	85.6%	83.8%
HCNN-4KM-GMM	94.1%	95.0%	94.6%	90.0%	92.7%	91.3%

In order to show the importance of the CNN_2^6 in our proposed HCNN architecture (Figure A3.3), we designed HCNN-4KM-GMM (Figure A4.1(c)) by removing the CNN_2^6 from our HCNN, and compared the classification results. As shown in Table A4.3, our proposed HCNN architecture achieved better classification results than HCNN-4KM-GMM, which proves that it is important to learn the correlation information among the 4 key moments before combining appearance and temporal features.

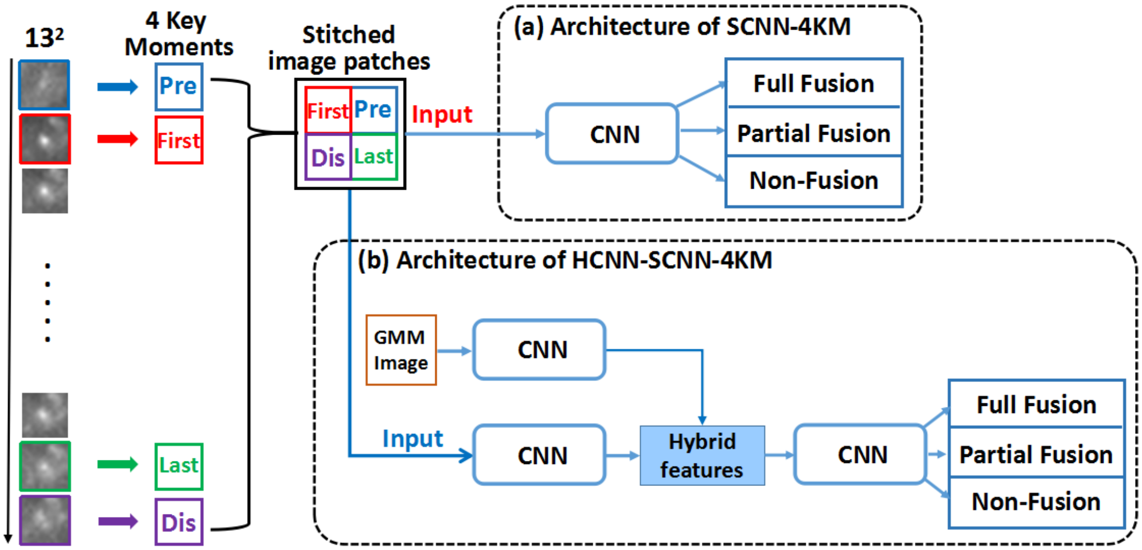


Figure A4.2. The architectures of the SCNN-4KM (a) and HCNN-SCNN-4KM (b).

Table A4.4. Comparison of different input formats of the appearance features on 9 challenging datasets, which include: SCNN-4KM (Figure A4.2(a)): we stitch the image patches from 4 key moments into an image, which will be the input to a CNN; HCNN-SCNN-4KM (Figure A4.2(b)): based on our HCNN, instead of using 4 CNNs, we use a CNN to learn the appearance features from stitched image patches for the classification.

	Full Fusion			Partial Fusion		
	Precision	Recall	F Score	Precision	Recall	F Score
Our Method	**95.2%**	**96.2%**	**95.7%**	**96.1%**	**96.7%**	**96.4%**
SCNN-4KM	93.7%	94.9%	94.3%	91.0%	93.2%	92.1%
HCNN-SCNN-4KM	94.8%	94.0%	94.4%	92.1%	91.7%	91.9%

4.4.2. Comparison of Alternative Appearance Feature Input Formats. In our HCNN architecture, we use 4 CNNs to learn the appearance features from the 4 key moments, where each CNN takes a single cropped image patch as input. In order to show the effectiveness of this design, we compared our HCNN architecture with SCNN-4KM (Figure A4.2(a)), which is one single CNN whose inputs are the stitched image patches from 4 key moments, and HCNN-SCNN-4KM (Figure A4.2(b)), which uses a CNN to learn appearance features from stitched image patches of 4 key moments and then combines with temporal features from GMM images for classification. As shown in Table A4.4, our proposed method achieved better classification results than SCNN-4KM and HCNN-SCNN-4KM, which validates the high-level appearance features extracted from 4 CNNs are more reliable for the fusion event classification task.

4.4.3. Comparison of Alternative Temporal Feature Input Formats. In our proposed HCNN architecture, each *GMM image* consists of 24 parameter sets which are organized as a 4×6 matrix pattern (Figure A3.2) to allow the HCNN to discover the hidden correlation among the parameter sets. In order to validate the effectiveness of our *GMM image* design, we compared our proposed 4×6 *GMM image* with the 24×1 *GMM image* (Figure A4.3(a)) and the 12×2 *GMM image* (Figure A4.3(b)). As shown in Table A4.5, our proposed HCNN architecture with 4×6 *GMM image* inputs achieved better classification results than HCNN-GMM(24×1) with 24×1 *GMM image* inputs and HCNN-GMM(12×2)

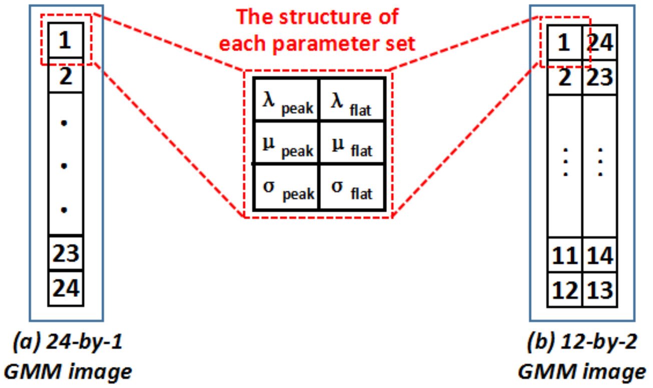


Figure A4.3. The structures of 24 × 1 *GMM image* (a) and 12 × 2 *GMM image* (b).

Table A4.5. Comparison of 2 alternative *GMM image* designs on 9 challenging datasets, which include the 24 × 1 *GMM image* (Figure A4.3(a)) in HCNN-GMM(24 × 1) and the 12 × 2 *GMM image* (Figure A4.3(b)) in HCNN-GMM(12 × 2).

	Full Fusion			Partial Fusion		
	Precision	Recall	F Score	Precision	Recall	F Score
Our Method	**95.2%**	**96.2%**	**95.7%**	**96.1%**	**96.7%**	**96.4%**
HCNN-GMM(24 × 1)	91.3%	94.0%	92.6%	92.3%	93.0%	92.7%
HCNN-GMM(12 × 2)	93.3%	93.5%	93.4%	92.3%	94.3%	93.3%

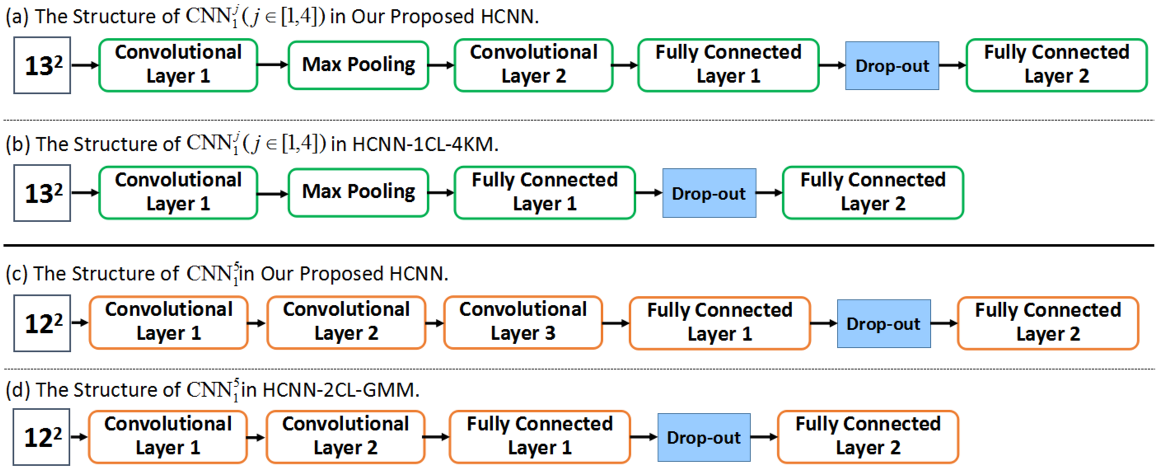
with 12 × 2 *GMM image* inputs. It proves that the 4 × 6 matrix pattern *GMM image*, which contains many 4- or 8-connected neighborhood relationships, can provide comprehensive information to reveal the unique pattern of the fusion event.

4.4.4. Comparison of Alternative CNN Designs. To validate the effectiveness of the individual CNNs in our proposed HCNN architecture, we tested different number of Convolutional Layers and Fully Connected Layers and compared with our proposed HCNN. Since it is unpractical to test all possible CNN structures, we only tested some reasonable CNN designs in this work.

In our proposed HCNN architecture, the structure of CNN_1^j ($j \in [1,4]$) has 2 Convolutional Layers (Figure A4.4(a)) and the structure of CNN_1^5 has 3 Convolutional Layers (Figure A4.4(c)). We designed HCNN-1CL-4KM (Figure A4.4(b)) by setting only

Table A4.6. Comparison of 4 alternative CNN designs in our proposed HCNN architecture on 9 challenging datasets, which include: HCNN-1CL-4KM, HCNN-2CL-GMM, HCNN-3FCL and HCNN-1FCL. These architectures are described in Section 4.4.4 in details.

	Full Fusion			Partial Fusion		
	Precision	Recall	F Score	Precision	Recall	F Score
Our Method	**95.2%**	**96.2%**	**95.7%**	**96.1%**	**96.7%**	**96.4%**
HCNN-1CL-4KM	86.0%	84.1%	85.0%	82.9%	85.4%	84.1%
HCNN-2CL-GMM	82.7%	80.2%	81.4%	81.4%	80.3%	80.9%
HCNN-3FCL	94.8%	95.0%	94.9%	95.9%	96.3%	96.1%
HCNN-1FCL	91.0%	93.5%	92.2%	93.2%	95.2%	94.2%



(a) The Structure of $\mathrm{CNN}_1^j (j \in [1,4])$ in Our Proposed HCNN.

(b) The Structure of $\mathrm{CNN}_1^j (j \in [1,4])$ in HCNN-1CL-4KM.

(c) The Structure of CNN_1^5in Our Proposed HCNN.

(d) The Structure of CNN_1^5in HCNN-2CL-GMM.

Figure A4.4. (a) The CNN_1^j ($j \in [1,4]$) structure in our proposed HCNN; (b) The CNN_1^j ($j \in [1,4]$) structure in HCNN-1CL-4KM; (c) The CNN_1^5 structure in our proposed HCNN; (d) The CNN_1^5 structure in HCNN-2CL-GMM.

1 Convolutional Layer to the structure of CNN_1^j ($j \in [1,4]$) in our proposed HCNN, where the other CNNs in HCNN-1CL-4KM are exactly the same with the ones in our proposed HCNN. We also designed HCNN-2CL-GMM (Figure A4.4(d)) by setting only 2 Convolutional Layers to the structure of CNN_1^5 in our proposed HCNN, where the other CNNs in HCNN-2CL-GMM are exactly the same with the ones in our proposed HCNN. As shown in Table A4.6, our proposed HCNN architecture outperformed HCNN-1CL-4KM and HCNN-2CL-GMM, which validates the effectiveness of the CNN_1^j ($j \in [1,5]$) in our proposed HCNN architecture.

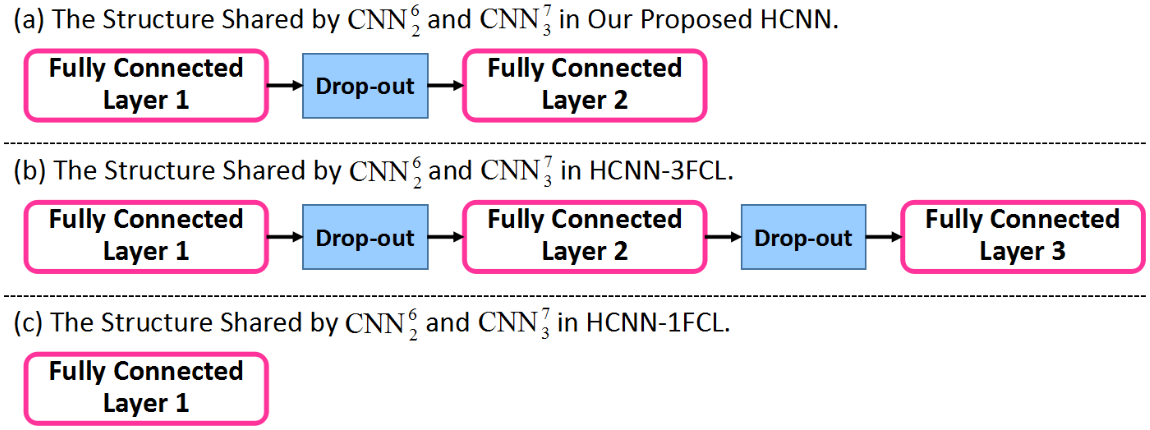

(a) The Structure Shared by CNN$_2^6$ and CNN$_3^7$ in Our Proposed HCNN.
Fully Connected Layer 1
Drop-out
Fully Connected Layer 2
(b) The Structure Shared by CNN$_2^6$ and CNN$_3^7$ in HCNN-3FCL.
Fully Connected Layer 1
Drop-out
Fully Connected Layer 2
Drop-out
Fully Connected Layer 3
(c) The Structure Shared by CNN$_2^6$ and CNN$_3^7$ in HCNN-1FCL.
Fully Connected Layer 1

Figure A4.5. (a) The structure shared by CNN_2^6 and CNN_3^7 in our proposed HCNN; (b) The structure shared by CNN_2^6 and CNN_3^7 in HCNN-3FCL; (c) The structure shared by CNN_2^6 and CNN_3^7 in HCNN-1FCL.

In our proposed HCNN architecture, the structure shared by CNN_2^6 and CNN_3^7 has 2 Fully Connected Layers (Figure A4.5(a)). We designed HCNN-3FCL (Figure A4.4(b)) by setting 3 Fully Connected Layers to the structure shared by CNN_2^6 and CNN_3^7, where the other settings in HCNN-3FCL are the same with our proposed HCNN. We also designed HCNN-1FCL (Figure A4.4(c)) by setting only 1 Fully Connected Layer to the structure shared by CNN_2^6 and CNN_3^7, where the other settings in HCNN-1FCL are the same with our proposed HCNN. As shown in Table A4.6, our proposed HCNN architecture achieved the best performance, which validates the effectiveness of the CNN_2^6 and CNN_3^7 in our proposed HCNN architecture.

4.5. DISCUSSION

According to the classification results of our proposed method, there are two main failure cases in our experiments. First, during our data collection, the Total Internal Reflection Fluorescent Microscope (TIRFM) sometimes was out of focus for several frames, as shown in Figure A4.8. Our proposed tracking method can still detect the image patches

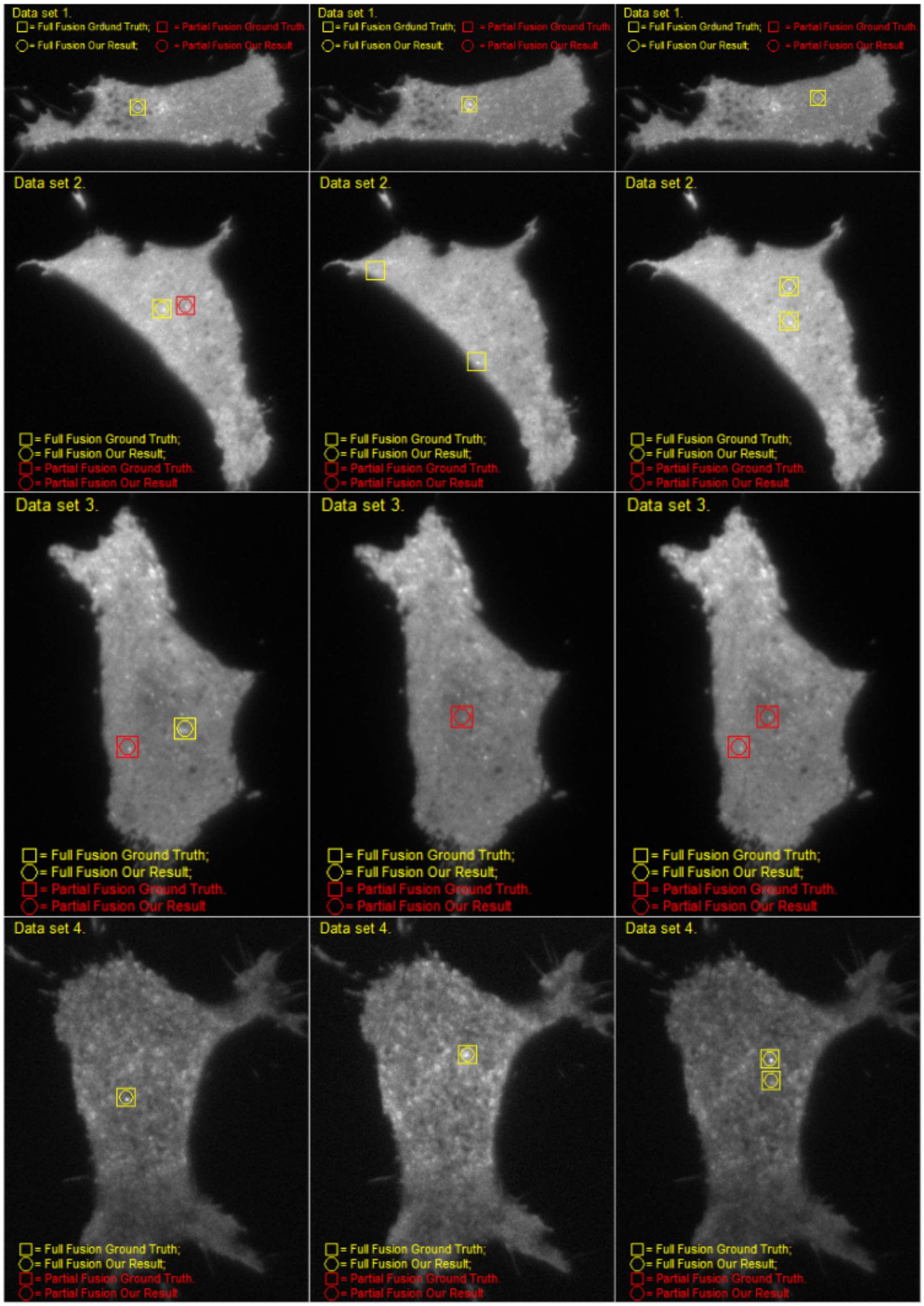

Data set 1.
□= Full Fusion Grdund Truth; □ = Partial Fusion Ground Truth
○= Full Fusion Our Result; ○ = Partial Fusion Our Result
Data set 1.
□= Full Fusion Grdund Truth; □ = Partial Fusion Ground Truth
○= Full Fusion Our Result; ○ = Partial Fusion Our Result
Data set 1.
□= Full Fusion Grdund Truth; □ = Partial Fusion Ground Truth
○= Full Fusion Our Result; ○ = Partial Fusion Our Result
Data set 2.
□= Full Fusion Ground Truth;
○= Full Fusion Our Result;
□= Partial Fusion Ground Truth.
○= Partial Fusion Our Result
Data set 2.
□= Full Fusion Ground Truth;
○= Full Fusion Our Result;
□= Partial Fusion Ground Truth.
○= Partial Fusion Our Result
Data set 2.
□= Full Fusion Ground Truth;
○= Full Fusion Our Result;
□= Partial Fusion Ground Truth.
○= Partial Fusion Our Result
Data set 3.
□= Full Fusion Ground Truth;
○= Full Fusion Our Result;
□= Partial Fusion Ground Truth.
○= Partial Fusion Our Result
Data set 3.
□= Full Fusion Ground Truth;
○= Full Fusion Our Result;
□= Partial Fusion Ground Truth.
○= Partial Fusion Our Result
Data set 3.
□= Full Fusion Ground Truth;
○= Full Fusion Our Result;
□= Partial Fusion Ground Truth.
○= Partial Fusion Our Result
Data set 4.
□= Full Fusion Ground Truth;
○= Full Fusion Our Result;
□= Partial Fusion Ground Truth.
○= Partial Fusion Our Result
Data set 4.
□= Full Fusion Ground Truth;
○= Full Fusion Our Result;
□= Partial Fusion Ground Truth.
○= Partial Fusion Our Result
Data set 4.
□= Full Fusion Ground Truth;
○= Full Fusion Our Result;
□= Partial Fusion Ground Truth.
○= Partial Fusion Our Result

Figure A4.6. Fusion event classification samples of dataset 1, 2, 3, 4 (yellow : full fusion; red: partial fusion; square: ground truth; circle: our result).

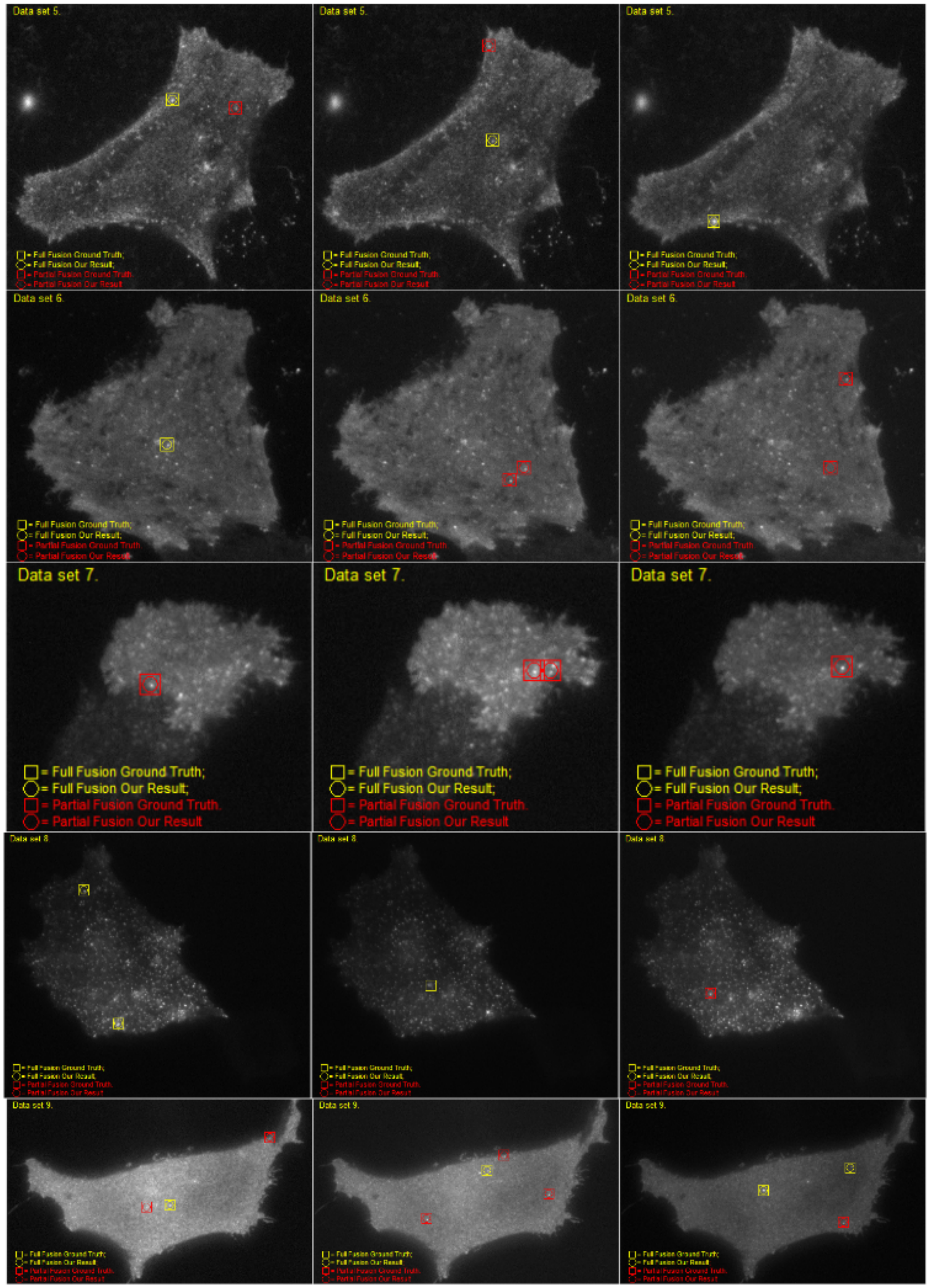

Data set 5.
□ = Full Fusion Ground Truth;
○ = Full Fusion Our Result;
□ = Partial Fusion Ground Truth.
○ = Partial Fusion Our Result
Data set 6.
Data set 7.
Data set 8
Data set 9.

Figure A4.7. Fusion event classification samples of dataset 5, 6, 7, 8, 9 (yellow : full fusion; red: partial fusion; square: ground truth; circle: our result).

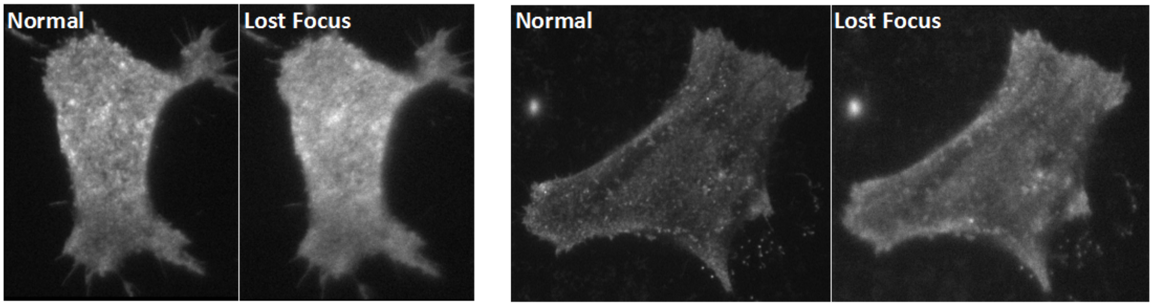


Figure A4.8. The TIRFM image samples which are affected by out-of-focus.

while the TIRFM is out of focus, but the intensity variation pattern of the fusion event is largely interfered by the out-of-focus problem, which misleads the HCNN to make a wrong classification. Second, some fusion events have extremely short lifetimes which are as short as 2 frames. For the short event process, the time-series intensity variation information from Gaussian fitting and the patches from the key moments are not very informative for the classification. Refining our current TIRFM hardware and increasing the image acquisition rate will be our future work to overcome the current drawbacks.

5. CONCLUSION

Accurately detecting and classifying vesicle-plasma membrane fusion events from TIRFM images is an essential research problem on cellular trafficking processes. In this paper, we proposed a novel Hierarchical Convolutional Neural Network (HCNN) based application to solve the fusion event detection and classification task. An adaptive detection & tracking method is developed to extract fusion event candidates and their time-series intensity variation information. By using the time-series intensity variation pattern introduced by Gaussian Mixture Models and the appearances in 4 key moments of the process of a fusion event, a HCNN architecture is proposed to classify fusion event candidates into three classes: full fusion, partial fusion and non-fusion. Our method showed its competitive performance and outperformed our previous work, two state-of-the-arts and eleven alternative neural network architectures on nine challenging datasets with low signal to noise ratio and frequent background fluctuations.

APPENDIX C.

AUTOMATED VESICLE FUSION DETECTION USING CONVOLUTIONAL NEURAL NETWORKS

1. INTRODUCTION AND RELATED WORKS

Vesicle exocytosis is an essential cellular trafficking process, by which materials (e.g., transporters, receptors and proteins) are transported from one membrane-bounded organelle to another or to the plasma membrane for growth and secretion. The analysis of these processes can provide deep insights on the cellular behavior in the diseased status Leney and Tavare (2009b)Bornemann *et al.* (1992b). The fusion interaction between vesicles and the cell membrane, which is able to be observed by using Total Internal Reflection Fluorescence Microscopy (TIRFM)Schneckenburger (2005b)Axelrod (1981b), can be represented in 2 momentous stages (Figure A1.1). In stage 1, vesicles are invisible in the pre-appearance frame, and then suddenly appear in the first-appearance frame as bright fluorescent circle spots. In stage 2, after halting for several frames, vesicles will either fuse on the cell membrane with a visible "halo" (full fusion events), or depart from the cell membrane with the circular shape (partial fusion events), which can be observed in the last appearance frame, respectively. Finally, vesicles under the full or partial fusion event will disappear in the disappearance frame. As the moving trajectory of a vesicle during the fusion process is almost perpendicular to the cell membrane, the vesicle fusion event projected onto the membrane surface (i.e., the image plane in TIRFM) has minute spatial displacement.

It is impractical to manually analyze TIRFM image sequences that typically consist of thousands of frames with hundreds of vesicles. Therefore, developing computational algorithms to automatically extract vesicle fusion information in TIRFM image sequences is badly needed to aid the quantitative study on the intercellular behavior. Image processing methods have been proposed to detect fusion events Bai *et al.* (2007b)Huang *et al.* (2007b)Basset *et al.* (2014b)Basset *et al.* (2015b). Individual vesicles in each frame are segmented by analyzing local gray scale distributions, then full and partial fusion events are

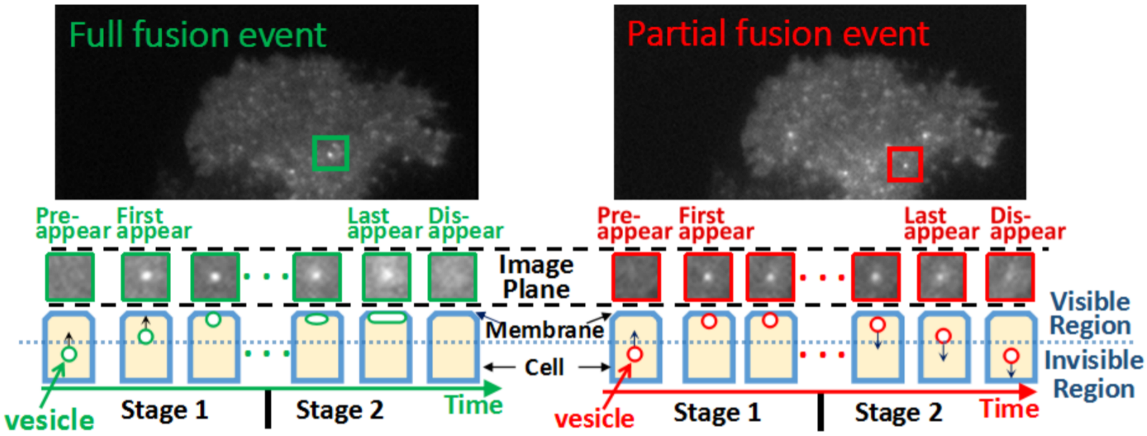


Figure A1.1. The 2 momentous stages of vesicle fusions and the related 4 key frames. Here are two real TIRFM images with a full fusion event (left) and a partial fusion event (right), respectively. During the fusion events, vesicles exhibit different patterns of appearance, brightness and shape in images.

classified by a pixel intensity threshold. But these methods are sensitive to the variation of intensity profiles (shown in Figure A2.1(c)). Based on both temporal and spatial features, Vallotton et al. Vallotton *et al.* (2007b) proposed a filter matching method, which is able to identify the fusion events with high correlation to a standard fusion event. However, due to the frequent background intensity fluctuation (shown in Figure A2.1(d,e,f)) introduced by the TIRFM system and intercellular activities, it is difficult to build a template that is representative for all fusion events. In order to enhance the tolerance to the variations of fusion events and the unpredictable noise interferences, some learning based methods were developed in recent years. Based on backpropagation neural network, Dosset et al. Dosset *et al.* (2016) developed an automatic method to detect fusion events by using a temporal sliding window. Li et al. Li (2015b) first applied a Gaussian Mixture Model (GMM) to fit on each individual fusion event, then a classifier was learned from the estimated parameters of GMMs to classify fusion events. However, the fixed temporal sliding window used in these methods may lose the critical information of fusion events with long duration.

2. CHALLENGES AND OUR PROPOSAL

Figure A2.1(a,b,c) show a few vesicle fusion event samples, from which we can observe some characteristics of vesicle fusion events regarding to their patterns of movement, shapes and intensities. However, it is challenging for automated image processing methods to distinguish vesicle fusion events from the large number of similar bright spots in TIRFM images. For instance, the circular background intensity fluctuation (Figure A2.1(d)) is similar to the vesicle fusion event. Some moving bright spots, which temporarily stay immobile near the cell membrane for several frames (Figure A2.1(e,f)), can be mistakenly considered as vesicle fusion events. In this paper, we explore both appearance features and temporal cues to detect and classify fusion events. Instead of a brute-force scanning on the input image sequence to detect fusion events, we extract fusion event candidate patch sequences to improve the detection efficiency. Then, we propose to build an event image that mosaics the critical frames of the candidate patch sequence into a single image. In addition to the visual appearance features in individual frames, the event image also embeds the temporal correlation among the critical frames into a single-image joint representation, which is used as the input to Convolutional Neural Networks (CNNs) Krizhevsky *et al.* (2012b). According to different lengths of the candidate patch sequences, adaptable formats of event images and their corresponding CNN architectures are designed to classify the candidate patch sequence into three classes: full fusion event, partial fusion event and non-fusion event.

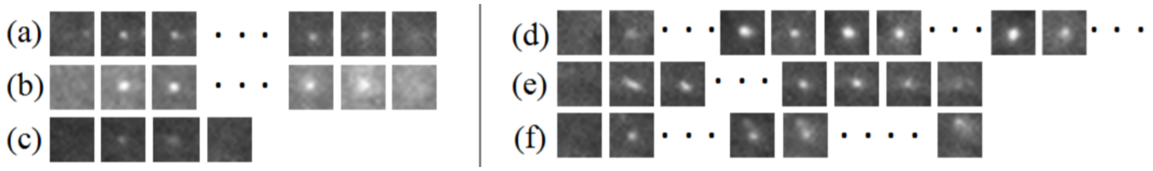


Figure A2.1. (a) A typical partial fusion event; (b) A typical full fusion event; (c) A short full fusion event is characterized by its halo; (d) A bright circular object caused by the background intensity fluctuation; (e) A moving bright spot, which only moves in the first several frames then stays immobile, is similar to a partial fusion event when it stops moving; (f) A background fluctuation, which is really similar to standard full fusion event in the early stage, then gradually moves out of the field of view.

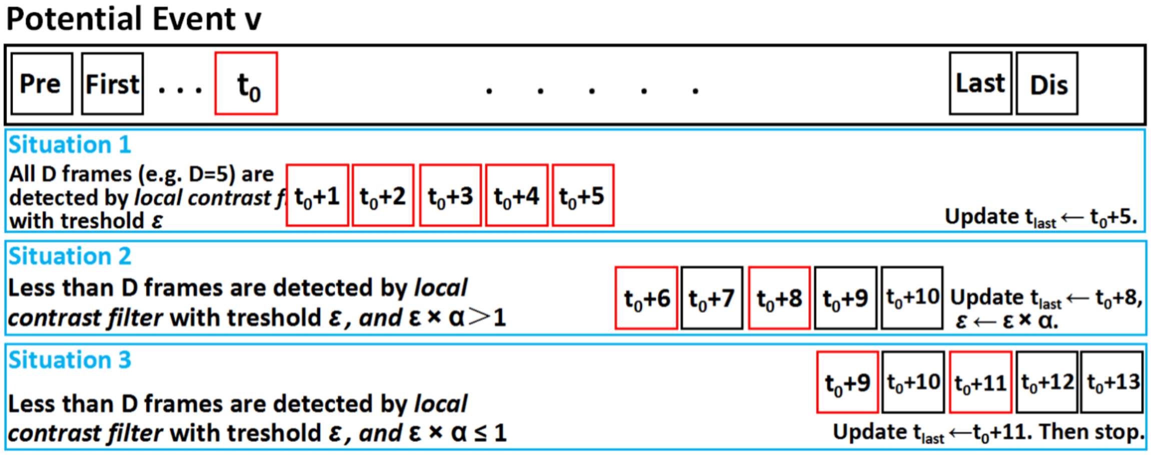

Potential Event v
Pre
First
. . .
t_0
Last
Dis
Situation 1
All D frames (e.g. D=5) are detected by *local contrast f* with treshold *ε*
t_0+1
t_0+2
t_0+3
t_0+4
t_0+5
Update $t_{last} \leftarrow t_0+5$.
Situation 2
Less than D frames are detected by *local contrast filter* with treshold *ε, and* ε × α > 1
t_0+6
t_0+7
t_0+8
t_0+9
t_0+10
Update $t_{last} \leftarrow t_0+8$, $ε \leftarrow ε \times α$.
Situation 3
Less than D frames are detected by *local contrast filter* with treshold *ε, and* ε × α ≤ 1
t_0+9
t_0+10
t_0+11
t_0+12
t_0+13
Update $t_{last} \leftarrow t_0+11$. Then stop.

Figure A2.2. An example to search the candidate patch sequence *S* in the forward temporal direction.

3. EXTRACT CANDIDATE PATCH SEQUENCES

As observed in the previous works Bai *et al.* (2007b)Xu *et al.* (2011b)Wu *et al.* (2015b), the vesicle fusion event appears to be a bright immobile circular spot, whose local contrast between its center and surrounding medium gradually decreases when the event disappears. Thus, we leverage the local spatial contrast to extract candidate patches in each frame, and then track them in the video sequence for the later classification, which has much better efficiency than exhaustively scanning the video volumes using spatiotemporal filters. Given the image I at time t_0, we compute the local contrast at each pixel location (x, y) as

$$f(x, y) = \frac{(n^2 - 1)I_{x,y}}{\sum_{(i,j)} I_{i,j}}, \tag{A3.1}$$

where (i, j) represents pixels in the n-by-n neighborhood around (x, y). Pixel (x, y) is possible to belong to a fusion event if$f(x, y)$ is larger than a threshold ϵ. Around a potential fusion event, there might be many pixels with their local contrast larger than the threshold. We find the pixel (x^*, y^*) with the local maximum of local contrast as the center of the potential fusion event and crop an n-by-n image patch around it. Since we use fixed size patches, we only need to record the coordinates of the patch center into the fusion event candidate patch sequence, which is denoted as $S = \left\{x_t^*, y_t^* | t \in [t_{first}, t_{last}\right\}]$ where t_{first} and t_{last} denote the first and last frame index of the patch sequence, respectively. At the beginning, $t_{first} = t_{last} = t_0$.

Then, we develop an iterative searching process to find the first-appearance frame and the last-appearance frame of a potential fusion event and every patch center within this time window. We use Figure A2.2 to illustrate the search in the forward direction to find the the last-appearance frame (the search in the backward direction to find the first-appearance frame is similar). During each iteration, we search the last-appearance frame in a sliding temporal window of D frames. Three situations are considered during the iterative search:

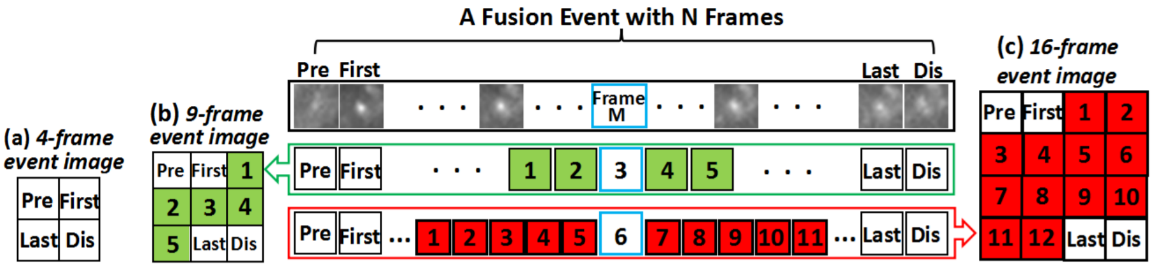


Figure A3.1. Build event image for the CNNs.

Situation 1, if the maximums of the local contrast in all D frames around location $(x^*_{t_{last}}, y^*_{t_{last}})$ are larger than ϵm then we update $S = \{x^*_t, y^*_t | t \in [t_{first}, t_{last}\}]$ by setting $t_{last} \leftarrow t_{last} + D$ and finding the patch centers (x^*, y^*) in the D frames which are the maximums of the local contrast.

Situation 2, if not all of the maximums of the local contrast in D frames around location $(x^*_{t_{last}}, y^*_{t_{last}})$ are larger than ϵ, while $\epsilon \times \alpha > 1$ (α is a decay rate on the threshold), we update t_{last} as the last frame within the D frame whose maximal local contrast is larger than ϵ and the patch centers are updated accordingly. The threshold is updated as $\epsilon \leftarrow \epsilon \times \alpha$.

Situation 3, if not all of the maximums of the local contrast in D frames around location $(x^*_{t_{last}}, y^*_{t_{last}})$ are are larger than ϵ and $\epsilon \times \alpha \leq 1$, we update the patch sequence similar to situation 2, then we stop the iteration.

By applying this iterative searching algorithm to the TIRFM image sequence, we can obtain potential fusion events in the format of candidate patch sequences, each of which records the coordinates of the patch center from the first-appearance frame to the last-appearance frame. For each potential fusion event, we compute the pairwise Euclidean distance between each consecutive pair of patch centers within the candidate patch sequence. If any of these distances is larger than the neighborhood size n, this candidate patch sequence is highly possible to be a non-fusion event cause by a moving object from the background, and we remove it from the candidate list.

In the experiment, we choose the following parameter setting: neighborhood size $n = 13$, sliding temporal window length $D = 5$, the initial threshold for local contrast $\epsilon = 1.3$ and the threshold decay rate $\alpha = 0.95$.

4. EVENT IMAGE AND CNN ARCHITECTURE

In this section, we propose an event image to mosaic image patches in the candidate sequence into a single image as the input to a Convolutional Neural Networks (CNNs). The event image contains both the visual appearance information of each individual patch and the visual correlation among different patches. The CNN automatically learns a comprehensive representation of temporal and spatial features from the event image for fusion event classification. By a series of parameterized layers, CNN maps each input event image into the probabilities of three classes: full fusion event, partial fusion event or non-fusion event.

The event image stitches critical patches from a candidate sequence into a single image by a specific order, which allows the CNN to discover not only the spatial and temporal information of the fusion event, but also the hidden correlation among its patches. Furthermore, we designed the event image as a square image so each patch has more chances to be neighbors of other patches. For example, given 16 patches, if we concatenate them into a 16-by-1 matrix pattern, there is no 4- or 8-connected neighborhood relationship among the patches. Rearranging the patches into a 8-by-2 matrix pattern increases the relationship a little. If we stitch the 16 patches into a 4-by-4 matrix pattern, a lot of 4- or 8-connected neighborhood relationship can be built among the patches.

Due to the large variation of the duration of vesicle fusion events, it is unpractical to design one fixed size of event image that fits all vesicle fusion events well. To distinguish the event images containing different numbers of image patches, we name an event image that contains k frames as k-frame event image (shown in Figure A3.1), where k is chosen to be a squared number to insure the event image be square sized.

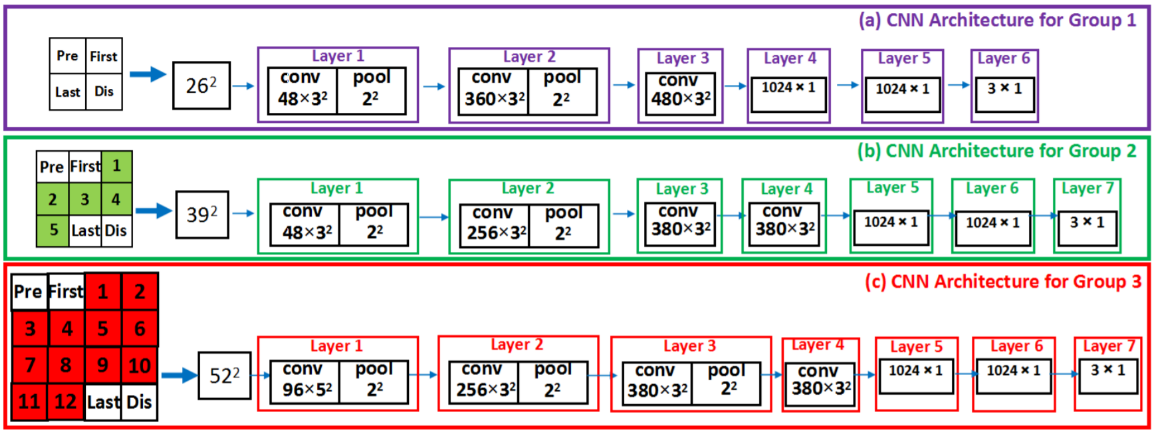


Figure A4.1. Our CNN architectures. (a) The CNN architecture for vesicle fusion events in Group 1, which accepts 4-frame event images with the size of 26 × 26 pixels; (b) The CNN architecture for vesicle fusion events in Group 2, which accepts 9-frame event images with the size of 39 × 39 pixels; (c) The CNN architecture for vesicle fusion events in Group 3, which accepts 16-frame event images with the size of 52 × 52 pixels; Note that, in all of these three architectures, each convolution process is followed by a rectified linear function (relu). Each max pooling is followed by a local normalization.

We categorize all vesicle fusion events into three groups based on their duration lengths. Group 1 contains vesicle fusion events having 4 to 6 frames, which takes image patches from the 4 key frames to construct 4-frame event images. Group 2 contains vesicle fusion events having 9 to 13 frames, which constructs 9-frame event images. Group 3 contains vesicle fusion events having 16 frames or more, which constructs 16-frame event images. For the vesicle fusion event with long duration in Group 2 or Group 3, we select image patches not only from the 4 key frames that represent its appearance and disappearance moments, but also from consecutive frames around the central frame M ($M = \lceil \frac{N}{2} \rceil$), which contain subtle characteristics of the variation pattern during the fusion process, as shown in Figure A3.1.

Then, the event images will be fed into the specific CNN architectures, as shown in Figure A4.1. In this paper, we adopt the MatConvNet Vedaldi and Lenc (2015) to design our CNN architectures. In the CNN architecture for Group 1, the first three layers are convolutional layers, where each of layer 1 and layer 2 is followed by a max-pooling that is used to extract local maximum in every 2×2 region. For the CNN for Group 2 and Group 3, we design four convolutional layers for each of them. Compared with Group 2, we design one more max-pooling following the third layer of CNNs in Group 3. In all of our CNNs, the last three layers are full connection layers. We minimize the softmax cost function at the last layer in each of these three CNNs, and use the back propagation to learn the parameters among the layers.

5. EXPERIMENT RESULTS

In this section, we validate the effectiveness of our framework in vesicle fusion classification on 9 challenging datasets.

5.1. DATASETS

We imaged different cell types with a variety of vesicle exocytosis in mammalian cells. These include constitutive exocytosis (transferrin receptor-pHluorin exocytosis in endothelial cells and 3T3-L1 adipocytes) and regulated exocytosis (VAMP2-pHluorin labeled insulin granule in MIN-6 cells and VAMP2-pHlurin labeled GLUT4 vesicle in 3T3-L1 adipocytes). In the experiments, 9 real TIRFM image sequences (examples are shown in Figure A5.1) were captured at 5 frame per second (fps), which consist of 15718 frames in total. Detailed specifications are summarized in Table A5.1. All datasets were well annotated by cell biologists working on vesicle trafficking analysis.

5.2. EXPERIMENT DESIGN & EVALUATION METRIC

We use the leave-one-out strategy to evaluate our method's performance, i.e., eight sequences are used for training while the last one for testing. In total, 9 leave-one-out experiments are performed on the datasets. The average performance on the 9 experiments in terms of precision, recall and F-score are used as the evaluation metrics.

Table A5.1. The specifications of our 9 datasets.

Dataset	1	2	3	4	5	6	7	8	9
# of Frames	2663	2661	2662	2662	579	1196	1665	428	1202
# of Full Fusions	118	169	31	132	48	16	19	76	193
# of Partial Fusions	28	64	56	6	10	16	76	11	191

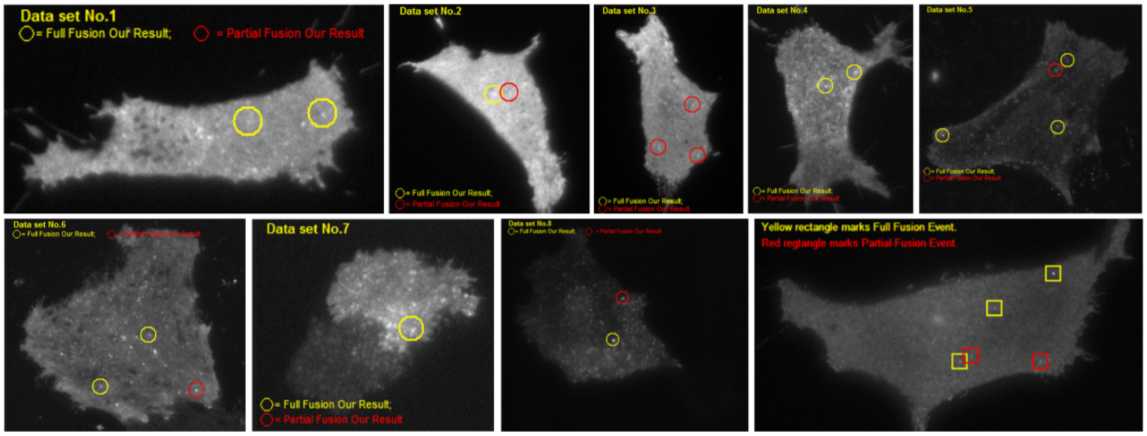

Data set No.1
○= Full Fusion Our Result;
○ = Partial Fusion Our Result
Data set No.2
○= Full Fusion Our Result;
○= Partial Fusion Our Result
Data set No.3
○= Full Fusion Our Result;
○= Partial Fusion Our Result
Data set No.4
○= Full Fusion Our Result;
○= Partial Fusion Our Result
Data set No.5
○= Full Fusion Our Result;
○= Partial Fusion Our Result
Data set No.6
○= Full Fusion Our Result;
○ = Partial Fusion Our Result
Data set No.7
○= Full Fusion Our Result;
○= Partial Fusion Our Result
Data set No.8
○= Full Fusion Our Result;
○ = Partial Fusion Our Result
Yellow rectangle marks Full Fusion Event.
Red regtangle marks Partial Fusion Event.

Figure A5.1. Examples of our detection on 9 datasets. (yellow: full fusion; red: partial fusion)

5.3. EFFECTIVENESS OF CANDIDATE PATCH SEQUENCE EXTRACTION

By using our proposed iterative searching algorithm, we obtain 4127 candidate patch sequences which contain all the 1260 vesicle fusion events (i.e., the recall is 100% and the precision is 1260/4127 = 30% from the detection step). Data augmentation techniques were applied on our positive training samples to provide enough training data.

5.4. COMPARISON WITH STATE-OF-THE-ARTS

We compare our algorithm with two state-of-the-arts: the learning-based Gaussian Mixture Model (GMM, Li (2015b)), and the intensity-based Single Gaussian Model (SGM, Bai *et al.* (2007b)). All parameters in Li (2015b) and Bai *et al.* (2007b) are optimized to ensure they can obtain their best performance in our TIRFM image sequences for fair com-

Table A5.2. The comparison of five methods on all datasets. GMMLi (2015b): Gaussian Mixture Model; SGMBai *et al.* (2007b): Single Gaussian Model; SCNN: Single-group CNN architecture; MCNN: Multi-channel CNN architecture.

Methods	Full Fusion			Partial Fusion		
	Precision	Recall	F-Score	Precision	Recall	F-Score
Ours	**95.0%**	**95.5%**	**95.2%**	**96.7%**	**96.1%**	**96.4%**
GMM Li (2015b)	77.0%	79.3%	78.1%	75.5%	76.0%	75.7%
SGM Bai *et al.* (2007b)	54.9%	64.7%	59.4%	64.6%	62.0%	63.0%
SCNN	93.7%	94.9%	94.3%	91.0%	93.2%	92.1%
MCNN	91.1%	91.0%	91.0%	88.2%	91.5%	89.8%

parisons. As shown in Table A5.2, compared with the GMM Li (2015b) with handcrafted features, our method achieves much better classification results for both the full and partial fusion events in 9 datasets, which validates that the proposed event image and the automatic feature selection by our CNN architectures have a more comprehensive representation of vesicle fusion events. Compared with the SGM Vedaldi and Lenc (2015) that only considers the spatial radius of the Gaussian fit to the bright blob, our method outperforms it by a large margin via using both the visual features and temporal cues hidden in the event image.

5.5. MULTI-GROUP CNN VS. SINGLE-GROUP CNN

We compared our multi-group CNN architectures with a Single-group CNN architecture (SCNN, i.e., for each fusion event, we only select image patches from the 4 key frames to construct the 4-frame event image for classification). SCNN uses the architecture in Figure A4.1(a). As shown in Table A5.2, the SCNN outperformed the two state-of-the-arts, while our method using three groups of event images and CNN architectures achieves even higher performance than SCNN.

5.6. MULTI-GROUP CNN VS. MULTI-CHANNEL CNN

Our proposed method is also compared with Multi-channel CNN architecture (MCNN, i.e., for every vesicle fusion event, we construct a 4-channel image by using its 4 key frames, as the input to a CNN). As shown in Table A5.2, both SCNN and our multi-group CNN architectures outperformed MCNN. We believe it is because the informative hidden correlation among the patches of the fusion event is incorporated into the CNN when event images are utilized.

6. CONCLUSION

In this paper, we first propose an iterative searching algorithm to extract patch sequences of potential fusion events, then design an event image to combine some informative patches of a candidate event into a single-image representation. According to different formats of event images, three specific Convolutional Neural Networks (CNNs) are designed to comprehensively learn the subtle characteristics of vesicle fusion events with different durations. All the potential events are classified by our CNNs into full-, partial-, or non-fusion events. Compared on 9 challenging datasets, our method showed very competitive performance and outperformed two state-of-the-arts.

REFERENCES

Axelrod, D., 'Cell-substrate contacts illuminated by total internal reflection fluorescence.' The Journal of cell biology, 1981a, **89**(1), pp. 141–145.

Axelrod, D., 'Cell-substrate contacts illuminated by total internal reflection fluorescence.' The Journal of cell biology, 1981b, **89**(1), pp. 141–145.

Baccouche, M., Mamalet, F., Wolf, C., Garcia, C., and Baskurt, A., 'Sequential deep learning for human action recognition,' in 'International workshop on human behavior understanding,' Springer, 2011 pp. 29–39.

Bahdanau, D., Cho, K., and Bengio, Y., 'Neural machine translation by jointly learning to align and translate,' arXiv preprint arXiv:1409.0473, 2014.

Bai, L., Wang, Y., Fan, J., Chen, Y., Ji, W., Qu, A., Xu, P., James, D. E., and Xu, T., 'Dissecting multiple steps of glut4 trafficking and identifying the sites of insulin action,' Cell metabolism, 2007a, **5**(1), pp. 47–57.

Bai, L., Wang, Y., Fan, J., Chen, Y., Ji, W., Qu, A., Xu, P., James, D. E., and Xu, T., 'Dissecting multiple steps of glut4 trafficking and identifying the sites of insulin action,' Cell metabolism, 2007b, **5**(1), pp. 47–57.

Basset, A., Bouthemy, P., Boulanger, J., Salamero, J., and Kervrann, C., 'Localization and classification of membrane dynamics in tirf microscopy image sequences,' in '2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI),' IEEE, 2014a pp. 830–833.

Basset, A., Bouthemy, P., Boulanger, J., Salamero, J., and Kervrann, C., 'Localization and classification of membrane dynamics in tirf microscopy image sequences,' in '2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI),' IEEE, 2014b pp. 830–833.

Basset, A., Bouthemy, P., Boulanger, J., Waharte, F., Kervrann, C., and Salamero, J., 'Detection and estimation of membrane diffusion during exocytosis in tirfm image sequences,' in '2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI),' IEEE, 2015a pp. 695–698.

Basset, A., Bouthemy, P., Boulanger, J., Waharte, F., Kervrann, C., and Salamero, J., 'Detection and estimation of membrane diffusion during exocytosis in tirfm image sequences,' in '2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI),' IEEE, 2015b pp. 695–698.

Bengio, Y., 'Deep learning of representations for unsupervised and transfer learning,' in 'Proceedings of ICML workshop on unsupervised and transfer learning,' 2012 pp. 17–36.

Beran, R., ‘Minimum hellinger distance estimates for parametric models,’ The Annals of Statistics, 1977, pp. 445–463.

Berger, L., Mirmehdi, M., Reed, S., and Tavaré, J., ‘A diffusion model for detecting and classifying vesicle fusion and undocking events,’ in ‘International Conference on Medical Image Computing and Computer-Assisted Intervention,’ Springer, 2012 pp. 329–336.

Bornemann, A., Ploug, T., and Schmalbruch, H., ‘Subcellular localization of glut4 in nonstimulated and insulin-stimulated soleus muscle of rat,’ Diabetes, 1992a, **41**(2), pp. 215–221.

Bornemann, A., Ploug, T., and Schmalbruch, H., ‘Subcellular localization of glut4 in nonstimulated and insulin-stimulated soleus muscle of rat,’ Diabetes, 1992b, **41**(2), pp. 215–221.

Boyer, J. S., ‘Plant productivity and environment,’ Science, 1982, **218**(4571), pp. 443–448.

Bui, T. D., Shin, J., and Moon, T., ‘3d densely convolutional networks for volumetric segmentation,’ arXiv preprint arXiv:1709.03199, 2017.

Chen, H., Qi, X., Yu, L., and Heng, P.-A., ‘Dcan: Deep contour-aware networks for accurate gland segmentation,’ Computer Vision and Pattern Recognition, 2016.

Chollet, F., ‘Keras,’ in ‘URL:https://github.com/fchollet/keras, GitHub repository,,’ 2015 .

Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., and Ronneberger, O., ‘3d u-net: Learning dense volumetric segmentation from sparse annotation,’ arXiv preprint arXiv:1709.03199, 2017.

Danson, F., Steven, M., Malthus, T., and Clark, J., ‘High-spectral resolution data for determining leaf water content,’ International Journal of Remote Sensing, 1992, **13**(3), pp. 461–470.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L., ‘Imagenet: A large-scale hierarchical image database,’ in ‘2009 IEEE conference on computer vision and pattern recognition,’ Ieee, 2009 pp. 248–255.

DeVries, T. and Taylor, G. W., ‘Improved regularization of convolutional neural networks with cutout,’ arXiv preprint arXiv:1708.04552, 2017.

Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell, T., ‘Decaf: A deep convolutional activation feature for generic visual recognition,’ in ‘International conference on machine learning,’ 2014 pp. 647–655.

Dosset, P., Rassam, P., Fernandez, L., Espenel, C., Rubinstein, E., Margeat, E., and Milhiet, P.-E., ‘Automatic detection of diffusion modes within biological membranes using back-propagation neural network,’ BMC bioinformatics, 2016, **17**(1), p. 197.

Dutt Jain, S. and Grauman, K., ‘Active image segmentation propagation,’ Computer Vision and Pattern Recognition, 2016, p. 2864–2873.

Earl, H. J. and Davis, R. F., ‘Effect of drought stress on leaf and whole canopy radiation use efficiency and yield of maize,’ Agronomy journal, 2003, **95**(3), pp. 688–696.

Feng, X., Qin, B., and Liu, T., ‘A language-independent neural network for event detection,’ Science China Information Sciences, 2018, **61**(9), p. 092106.

Fiorani, F., Rascher, U., Jahnke, S., and Schurr, U., ‘Imaging plants dynamics in heterogenic environments,’ Current opinion in biotechnology, 2012, **23**(2), pp. 227–235.

Fischler, M. A. and Bolles, R. C., ‘Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,’ Communications of the ACM, 1981, **24**(6), pp. 381–395.

Fu, Y., Wang, X., Wei, Y., and Huang, T., ‘Sta: Spatial-temporal attention for large-scale video-based person re-identification,’ The Association for the Advancement of Artificial Intelligence, 2019.

Gers, F. A., Schmidhuber, J., and Cummins, F., ‘Learning to forget: Continual prediction with lstm,’ 1999.

Godinez, W. J., Lampe, M., Koch, P., Eils, R., Muller, B., and Rohr, K., ‘Identifying virus-cell fusion in two-channel fluorescence microscopy image sequences based on a layered probabilistic approach,’ IEEE transactions on medical imaging, 2012, **31**(9), pp. 1786–1808.

Godinez, W. J., Lampe, M., Worz, S., Eils, R., Muller, B., and Rohr, K., ‘Identifying fusion events in fluorescence microscopy images,’ in ‘2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro,’ IEEE, 2009 pp. 1170–1173.

Gong, S., Cristani, M., Yan, S., and Loy, C. C., ‘Person reidentification,’ Springer, 2013, pp. 152–159.

Graham, S., Chen, H., Gamper, J., Dou, Q., Heng, P.-A., Snead, D., Tsang, Y. W., and Rajpoot, N., ‘Mild-net: Minimal information loss dilated network for gland instance segmentation in colon histology images,’ Medical Image Analysis, 2019, pp. 487–500.

Haboudane, D., Tremblay, N., Miller, J. R., and Vigneault, P., ‘Remote estimation of crop chlorophyll content using spectral indices derived from hyperspectral data,’ IEEE Transactions on Geoscience and Remote Sensing, 2008, **46**(2), pp. 423–437.

Hasegawa, P. M., Bressan, R. A., Zhu, J.-K., and Bohnert, H. J., ‘Plant cellular and molecular responses to high salinity,’ Annual review of plant biology, 2000, **51**(1), pp. 463–499.

Hermans, A., Beyer, L., and Leibe, B., 'In defense of the triplet loss for person re-identification,' arXiv preprint arXiv:1703.07737, 2017.

Hirzer, M., Beleznai, C., Roth, P. M., and Bischof, H., 'Person re-identification by descriptive and discriminative classification,' Scandinavian conference on Image analysis, 2011, pp. 91–102.

Hochreiter, S., 'The vanishing gradient problem during learning recurrent neural nets and problem solutions,' International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 1998, **6**(02), pp. 107–116.

Hong, S., Noh, H., and Han, B., 'Decoupled deep neural network for semi-supervised semantic segmentation,' ????

Hou, J. C. and Pessin, J. E., 'Ins (endocytosis) and outs (exocytosis) of glut4 trafficking,' Current opinion in cell biology, 2007, **19**(4), pp. 466–473.

Hou, R., Ma, B., Chang, H., Gu, X., Shan, S., and Chen, X., 'Vrstc: Occlusion-free video person re-identification,' Computer Vision and Pattern Recognition, 2019, pp. 7183–7192.

Hu, B., Lu, Z., Li, H., and Chen, Q., 'Convolutional neural network architectures for matching natural language sentences,' in 'Advances in neural information processing systems,' 2014 pp. 2042–2050.

Hu, J., Shen, L., and Sun, G., 'Squeeze-and-excitation networks,' Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.

Huang, S., Lifshitz, L. M., Jones, C., Bellve, K. D., Standley, C., Fonseca, S., Corvera, S., Fogarty, K. E., and Czech, M. P., 'Insulin stimulates membrane fusion and glut4 accumulation in clathrin coats on adipocyte plasma membranes,' Molecular and cellular biology, 2007a, **27**(9), pp. 3456–3469.

Huang, S., Lifshitz, L. M., Jones, C., Bellve, K. D., Standley, C., Fonseca, S., Corvera, S., Fogarty, K. E., and Czech, M. P., 'Insulin stimulates membrane fusion and glut4 accumulation in clathrin coats on adipocyte plasma membranes,' Molecular and cellular biology, 2007b, **27**(9), pp. 3456–3469.

Hunt Jr, E. R. and Rock, B. N., 'Detection of changes in leaf water content using near- and middle-infrared reflectances,' Remote sensing of environment, 1989, **30**(1), pp. 43–54.

Israelsen, O. W. and West, F. L. R., *Water-holding capacity of irrigated soils*, 183, Utah Agricultural College Experiment Station, 1922.

Jahn, R. and Fasshauer, D., 'Molecular machines governing exocytosis of synaptic vesicles,' Nature, 2012, **490**(7419), pp. 201–207.

Joshi, A. J., Porikli, F., and Papanikolopoulos, N. P., 'Scalable active learning for multiclass image classification,' Pattern Analysis and Machine Intelligence, 2012, pp. 2259–2273.

Kamnitsas, K., Ledig, C., Newcombe, V. F., Simpson, J. P., Kane, A. D., Menon, D. K., Rueckert, D., and Glocker, B., 'Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation,' Medical Image Analysis, 2017, p. 61–78.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., and Fei-Fei, L., 'Large-scale video classification with convolutional neural networks,' in 'Proceedings of the IEEE conference on Computer Vision and Pattern Recognition,' 2014 pp. 1725–1732.

Khan, F. and Bremond, F., 'Multi-shot person re-identification using part appearance mixture,' Winter Conference on Application of Computer Vision, 2017, pp. 605–614.

Kim, Y., 'Convolutional neural networks for sentence classification,' arXiv preprint arXiv:1408.5882, 2014.

Krizhevsky, A., Sutskever, I., and Hinton, G. E., 'Imagenet classification with deep convolutional neural networks,' in 'Advances in neural information processing systems,' 2012a pp. 1097–1105.

Krizhevsky, A., Sutskever, I., and Hinton, G. E., 'Imagenet classification with deep convolutional neural networks,' in 'Advances in neural information processing systems,' 2012b pp. 1097–1105.

Lawrence, S., Giles, C. L., Tsoi, A. C., and Back, A. D., 'Face recognition: A convolutional neural-network approach,' IEEE transactions on neural networks, 1997, **8**(1), pp. 98–113.

Leney, S. E. and Tavare, J. M., 'The molecular basis of insulin-stimulated glucose uptake: signalling, trafficking and potential drug targets.' The Journal of endocrinology, 2009a, **203**(1), pp. 1–18.

Leney, S. E. and Tavare, J. M., 'The molecular basis of insulin-stimulated glucose uptake: signalling, trafficking and potential drug targets.' The Journal of endocrinology, 2009b, **203**(1), pp. 1–18.

Li, H., 'A gaussian mixture model for automated vesicle fusion detection and classification,' 2015a.

Li, H., 'A gaussian mixture model for automated vesicle fusion detection and classification,' 2015b.

Li, S., Bak, S., Carr, P., and Wang, X., 'Diversity regularized spatiotemporal attention for video-based person re-identification,' Computer Vision and Pattern Recognition, 2018a, pp. 369–378.

Li, W., Zhu, X. T., and Gong, S. G., 'Harmonious attention network for person re-identification,' Computer Vision and Pattern Recognition, 2018b.

Liao, F., Liang, M., Li, Z., Hu, X., and Song, S., 'Evaluate the malignancy of pulmonary nodules using the 3d deep leaky noisy-or network,' arXiv preprint arXiv:1711.0832, 2017.

Lin, Y., Zheng, L., Zheng, Z., Wu, Y., Hu, Z., Yan, C., and Yang, Y., 'Improving person re-identification by attribute and identity learning,' Pattern Recognition, 2019.

Lin, Z., Feng, M., d. Santos, C. N., Yu, M., Xiang, B., Zhou, B., and Bengio, Y., 'A structured self-attentive sentence embedding,' arXiv preprint arXiv:1703.03130, 2017.

Linsley, D., Shiebler, D., Eberhardt, S., and Serre, T., 'Learning what and where to attend,' arXiv preprint arXiv:1805.08819, 2018.

Liu, C., Wu, C., Wang, Y. F., and Chien, S., 'Spatially and temporally efficient non-local attention network for video-based person re-identification,' British Machine Vision Conference, 2019.

Liu, H., Jie, Z., Jayashree, K., Qi, M., Jiang, J., Yan, S., and Feng, J., 'Video-based person re-identification with accumulative motion context,' arXiv preprint arXiv:1701.00193, 2017.

Liu, J., Shahroudy, A., Xu, D., and Wang, G., 'Spatio-temporal lstm with trust gates for 3d human action recognition,' in 'European conference on computer vision,' Springer, 2016 pp. 816–833.

Long, J., Shelhamer, E., and Darrell, T., 'Fully convolutional networks for semantic segmentation,' Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.

Malenovskỳ, Z., Mishra, K. B., Zemek, F., Rascher, U., and Nedbal, L., 'Scientific and technical challenges in remote sensing of plant canopy reflectance and fluorescence,' Journal of experimental botany, 2009, **60**(11), pp. 2987–3004.

Mao, Y. and Yin, Z., 'A hierarchical convolutional neural network for mitosis detection in phase-contrast microscopy images,' in 'International Conference on Medical Image Computing and Computer-Assisted Intervention,' Springer, 2016 pp. 685–692.

Mao, Y., Yin, Z., and Schober, J., 'A deep convolutional neural network trained on representative samples for circulating tumor cell detection,' in '2016 IEEE Winter Conference on Applications of Computer Vision (WACV),' IEEE, 2016 pp. 1–6.

McLaughlin, N., del Rincon, J. M., and Miller, P., 'Joint detection and identification feature learning for person search,' Computer Vision and Pattern Recognition, 2016a, pp. 1325–1334.

McLaughlin, N., Rincon, J. M., and Miller, P., 'Recurrent convolutional network for video-based person re-identification,' Computer Vision and Pattern Recognition, 2016b, pp. 1325–1334.

Miesenböck, G., De Angelis, D. A., and Rothman, J. E., ‘Visualizing secretion and synaptic transmission with ph-sensitive green fluorescent proteins,’ Nature, 1998, **394**(6689), pp. 192–195.

Oda, M., Shimizu, N., Roth, H. R., Karasawa, K., Kitasaka, T., Misawa, K., Fujiwara, M., Rueckert, D., and Mori, K., ‘3d fcn feature driven regression forest-based pancreas localization and segmentation,’ Deep Learning in Medical Image Analysis and Multimodal Learning, 2017, p. 222–230.

Papandreou, G., Chen, L.-C., Murphy, K., and Yuille, A., ‘Weakly- and semi-supervised learning of a dcnn for semantic image segmentation,’ International Conference on Computer Vision, 2015, pp. 1742–1750.

Parascandolo, G., Huttunen, H., and Virtanen, T., ‘Recurrent neural networks for polyphonic sound event detection in real life recordings,’ in ‘2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),’ IEEE, 2016 pp. 6440–6444.

Rascher, U., Damm, A., van der Linden, S., Okujeni, A., Pieruschka, R., Schickling, A., and Hostert, P., ‘Sensing of photosynthetic activity of crops,’ in ‘Precision Crop Protection-the Challenge and Use of Heterogeneity,’ pp. 87–99, Springer, 2010.

Richards, R. A., Rebetzke, G. J., Watt, M., Condon, A. T., Spielmeyer, W., and Dolferus, R., ‘Breeding for improved water productivity in temperate cereals: phenotyping, quantitative trait loci, markers and the selection environment,’ Functional Plant Biology, 2010, **37**(2), pp. 85–97.

Rizzoli, S. O. and Jahn, R., ‘Kiss-and-run, collapse and ‘readily retrievable’vesicles,’ Traffic, 2007, **8**(9), pp. 1137–1144.

Römer, C., Wahabzada, M., Ballvora, A., Pinto, F., Rossini, M., Panigada, C., Behmann, J., Léon, J., Thurau, C., Bauckhage, C., *et al.*, ‘Early drought stress detection in cereals: simplex volume maximisation for hyperspectral image analysis,’ Functional Plant Biology, 2012, **39**(11), pp. 878–890.

Schneckenburger, H., ‘Total internal reflection fluorescence microscopy: technical innovations and novel applications,’ Current opinion in biotechnology, 2005a, **16**(1), pp. 13–18.

Schneckenburger, H., ‘Total internal reflection fluorescence microscopy: technical innovations and novel applications,’ Current opinion in biotechnology, 2005b, **16**(1), pp. 13–18.

Schuster, M. and Paliwal, K. K., ‘Bidirectional recurrent neural networks,’ IEEE transactions on Signal Processing, 1997, **45**(11), pp. 2673–2681.

Settles, B., ‘Active learning literature survey,’ Dept. Computer Science, University of Wisconson–Madison, Madison, WI, USA, Tech. Rep. 1648, 2009.

Simonyan, K. and Zisserman, A., ‘Very deep convolutional networks for large-scale image recognition,’ arXiv preprint arXiv:1409.1556, 2014.

Sirinukunwattana, K., Pluim, J. P., Chen, H., Qi, X., Heng, P.-A., Guo, Y. B., Wang, L. Y., Matuszewski, B. J., Bruni, E., and Sanchez, U., ‘Gland segmentation in colon histology images: the glas challenge contest,’ Medical Image Analysis, 2017, pp. 489–502.

Smith, M. B., Karatekin, E., Gohlke, A., Mizuno, H., Watanabe, N., and Vavylonis, D., ‘Interactive, computer-assisted tracking of speckle trajectories in fluorescence microscopy: application to actin polymerization and membrane fusion,’ Biophysical journal, 2011, **101**(7), pp. 1794–1804.

Somerville, C. and Briscoe, J., ‘Genetic engineering and water,’ 2001.

Song, C. F., Huang, Y., Ouyang, W. L., and Wang, L., ‘Mask-guided contrastive attention model for person re-identification,’ Computer Vision and Pattern Recognition, 2018, pp. 1179–1188.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R., ‘Dropout: a simple way to prevent neural networks from overfitting,’ The journal of machine learning research, 2014, **15**(1), pp. 1929–1958.

Su, C., Zhang, S., Xing, J., Gao, W., and Tian, Q., ‘Deep attributes driven multi-camera person re-identification,’ European Conference on Computer Vision, 2016.

SUTER, B. W., ‘The multilayer perceptron as an approximation to a bayes optimal discriminant function,’ IEEE Transactions on Neural Networks, 1990, **1**(4), p. 291.

Thenkabail, P. S., Smith, R. B., and De Pauw, E., ‘Hyperspectral vegetation indices and their relationships with agricultural crop characteristics,’ Remote sensing of Environment, 2000, **71**(2), pp. 158–182.

Tilling, A. K., O’Leary, G. J., Ferwerda, J. G., Jones, S. D., Fitzgerald, G. J., Rodriguez, D., and Belford, R., ‘Remote sensing of nitrogen and water stress in wheat,’ Field Crops Research, 2007, **104**(1-3), pp. 77–85.

Ustin, S. L. and Gamon, J. A., ‘Remote sensing of plant functional types,’ New Phytologist, 2010, **186**(4), pp. 795–816.

Vallotton, P., James, D. E., and Hughes, W. E., ‘Towards fully automated identification of vesicle-membrane fusion events in tirf microscopy,’ in ‘AIP Conference Proceedings,’ volume 952, American Institute of Physics, 2007a pp. 3–10.

Vallotton, P., James, D. E., and Hughes, W. E., ‘Towards fully automated identification of vesicle-membrane fusion events in tirf microscopy,’ in ‘AIP Conference Proceedings,’ volume 952, American Institute of Physics, 2007b pp. 3–10.

Vedaldi, A. and Lenc, K., 'Matconvnet: Convolutional neural networks for matlab,' in 'Proceedings of the 23rd ACM international conference on Multimedia,' 2015 pp. 689–692.

Virlet, N., Sabermanesh, K., Sadeghi-Tehran, P., and Hawkesford, M. J., 'Field scanalyzer: An automated robotic field phenotyping platform for detailed crop monitoring,' Functional Plant Biology, 2017, **44**(1), pp. 143–153.

Wang, L., Nie, D., Li, G., Puybareau, É., Dolz, J., Zhang, Q., Wang, F., Xia, J., Wu, Z., and Chen, J., 'Benchmark on automatic 6-month-old infant brain segmentation algorithms: the iseg-2017 challenge,' Transactions on Medical Imaging, 2019.

Wang, S. and Jiang, J., 'Learning natural language inference with lstm,' arXiv preprint arXiv:1512.08849, 2015.

Wang, T., Gong, S., Zhu, X., and Wang., S., 'Person re-identification by video ranking,' European Conference on Computer Vision, 2014, pp. 688–703.

Wang, X. L., Girshick, R., Gupta, A., and He, K. M., 'Deep residual learning for image recognition,' Computer Vision and Pattern Recognition, 2017a, pp. 770–778.

Wang, X. L., Girshick, R., Gupta, A., and He, K. M., 'Deep residual learning for image recognition,' Computer Vision and Pattern Recognition, 2017b, pp. 770–778.

Wang, X. L., Girshick, R., Gupta, A., and He, K. M., 'Non-local neural networks,' Computer Vision and Pattern Recognition, 2018, pp. 7794–7803.

Wen, T.-H., Gasic, M., Mrksic, N., Su, P.-H., Vandyke, D., and Young, S., 'Semantically conditioned lstm-based natural language generation for spoken dialogue systems,' arXiv preprint arXiv:1508.01745, 2015.

Wu, J., Xu, Y., Feng, Z., and Zheng, X., 'Automatically identifying fusion events between glut4 storage vesicles and the plasma membrane in tirf microscopy image sequences,' Computational and mathematical methods in medicine, 2015a, **2015**.

Wu, J., Xu, Y., Feng, Z., and Zheng, X., 'Automatically identifying fusion events between glut4 storage vesicles and the plasma membrane in tirf microscopy image sequences,' Computational and mathematical methods in medicine, 2015b, **2015**.

Xiao, H., Wei, Y., Liu, Y., Zhang, M., and Feng, J., 'Transferable semi-supervised semantic segmentation,' Association for the Advancement of Artificial Intelligence, 2018.

Xiao, Q. Q., Luo, H., and Zhang, C., 'Margin sample mining loss: a deep learning based method for person reidentification,' arXiv preprint arXiv:1710.00478, 2017.

Xiong, L., Wang, R.-G., Mao, G., and Koczan, J. M., 'Identification of drought tolerance determinants by genetic analysis of root response to drought stress and abscisic acid,' Plant physiology, 2006, **142**(3), pp. 1065–1074.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y., 'Show, attend and tell: Neural image caption generation with visual attention,' in 'International conference on machine learning,' 2015 pp. 2048–2057.

Xu, S. J., Cheng, Y., Gu, K., Yang, Y., Chang, S. Y., and Zhou, P., 'Jointly attentive spatial-temporal pooling networks for video-based person re-identification,' arXiv preprint, 2017.

Xu, Y., Nan, D., Fan, J., Bogan, J. S., and Toomre, D., 'Optogenetic activation reveals distinct roles of pip3 and akt in adipocyte insulin action,' Journal of cell science, 2016, **129**(10), pp. 2085–2095.

Xu, Y., Rubin, B. R., Orme, C. M., Karpikov, A., Yu, C., Bogan, J. S., and Toomre, D. K., 'Dual-mode of insulin action controls glut4 vesicle exocytosis,' Journal of Cell Biology, 2011a, **193**(4), pp. 643–653.

Xu, Y., Rubin, B. R., Orme, C. M., Karpikov, A., Yu, C., Bogan, J. S., and Toomre, D. K., 'Dual-mode of insulin action controls glut4 vesicle exocytosis,' Journal of Cell Biology, 2011b, **193**(4), pp. 643–653.

Yan, Y., Ni, B., Song, Z., Ma, C., Yan, Y., and Yang, X., 'Person re-identification via recurrent feature aggregation,' European Conference on Computer Vision, 2016, pp. 701–716.

Yang, L., Zhang, Y., Chen, J., Zhang, S., and Chen, D. Z., 'Suggestive annotation: a deep active learning framework for biomedical image segmentation,' Medical Image Computing and Computer Assisted Intervention, 2017, pp. 399–407.

Yilmaz, M. T., Hunt Jr, E. R., and Jackson, T. J., 'Remote sensing of vegetation water content from equivalent water thickness using satellite imagery,' Remote Sensing of Environment, 2008, **112**(5), pp. 2514–2522.

Yu, J., Holland, J. B., McMullen, M. D., and Buckler, E. S., 'Genetic design and statistical power of nested association mapping in maize,' Genetics, 2008, **178**(1), pp. 539–551.

Yuan, T., Lu, J., Zhang, J., Zhang, Y., and Chen, L., 'Spatiotemporal detection and analysis of exocytosis reveal fusion "hotspots" organized by the cytoskeleton in endocrine cells,' Biophysical journal, 2015, **108**(2), pp. 251–260.

Yue-Hei Ng, J., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., and Toderici, G., 'Beyond short snippets: Deep networks for video classification,' in 'Proceedings of the IEEE conference on computer vision and pattern recognition,' 2015 pp. 4694–4702.

Zeng, M. Y., Tian, C., and Wu, Z. M., 'Person re-identification with hierarchical deep learning feature and efficient xqda metric,' ACM Multimedia Conference, 2018, pp. 1838–1846.

Zhang, L., 'Active learning based on locally linear reconstruction,' Pattern Analysis and Machine Intelligence, 2011, pp. 2026–2038.

Zhao, Y., Shen, X., Jin, Z., Lu, H., and Hua, X., 'Attribute-driven feature disentangling and temporal aggregation for video person re-identification,' Computer Vision and Pattern Recognition, 2019, pp. 4913–4922.

Zheng, L., Bie, Z., Sun, Y., Wang, J., C. Su, S. W., and Tian, Q., 'Mars: A video benchmark for large-scale person re-identification,' European Conference on Computer Vision, 2016, pp. 868–884.

VITA

Dr. Haohan Li received his B.E. degree in July 2011 at the South China University of Technology. He received his M.S. degree in Computer Science in May 2015 and his Ph.D. degree in Computer Science in May 2020 from the Missouri University of Science and Technology. His major research areas included biomedical image analysis, microscopy image analysis, deep learning, and active learning. Particularly, he was interested in the research attention mechanism algorithm design and its related applications in computer vision.