CHARACTERIZING CONSTRUCTION EQUIPMENT ACTIVITIES IN LONG VIDEO SEQUENCES OF EARTHMOVING OPERATIONS VIA KINEMATIC FEATURES

BY

RUXIAO BAO

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Civil Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2015

Urbana, Illinois

Advisor:

Assistant Professor Mani Golparvar-Fard

ABSTRACT

This thesis presents a fast and scalable method for activity analysis of construction equipment involved in earthmoving operations from highly varying long-sequence videos obtained from fixed cameras. A common approach to characterize equipment activities consists of detecting and tracking the equipment within the video volume, recognizing interest points and describing them locally, followed by a bag-of-words representation for classifying activities. While successful results have been achieved in each aspect of detection, tracking, and activity recognition, the highly varying degree of intra-class variability in resources, occlusions and scene clutter, the difficulties in defining visually-distinct activities, together with long computational time have challenged scalability of current solutions. In this thesis, we present a new end-to-end automated method to recognize the equipment activities by simultaneously detecting and tracking features, and characterizing the spatial kinematics of features via a decision tree. The method is tested on an unprecedented dataset of 5hr-long real-world videos of interacting pairs of excavators and trucks. The Experimental results show that the method is capable of activity recognition with accuracy of 88.91% with a computational time less than 1-to-1 ratio for each video length. The benefits of the proposed method for root-cause assessment of performance deviations are discussed.

ACKNOWLEDGEMENTS

The author wishes to give profound gratitude to my advisor Professor Mani Golparvar-Fard for giving me the opportunity to be a part of this project and all guidance and support he has given me along the way. I have thoroughly enjoyed working with him and feel like I have learned a lot.

The author greatly appreciate the guidance from, and interaction with my collaborator at Computer Science department, in particular Dr. Mohammad Amin Sadeghi. It was a pleasure and an honor to work with him on this project. I would also like to thank to Professor Derek Hoiem, graduate student Jiayi Duan and Rongqi Gu, undergraduate student Minwoo Eom for their guidance and contribution.

The author wishes to also express sincere appreciation to Caterpillar Inc. and Caterpillar data innovation center at Research Park, Champaign, IL, for funding this research and providing all the video data for us.

TABLE OF CONTENTS

LIST OF SYMBOLS

n	Feature space dimension
$F(n)$	Feature space
S_b	Block size
S_c	Cell size
L_C	Number of convolutional layers of CNN
L_p	Number of pooling layers of CNN
y_I	Category of SVM classification
x_i	Feature points of SVM classification
b_i	Constant of SVM
d_i	Feature space dimension
w	Weight
ST	State transition model
M	Measurement model
PN	Process noise
MN	Measurement noise
i	Part index, i = 1, 2, 3 stands for Parts “Arm Tip”, “Bucket”, and “Body”

j	Activity index, j = 1~4 stands for atomic activities "Moving" "Idling", "Swinging", "Digging/dumping"
x_i	X coordinate in the frame of part i
y_i	Y coordinate in the frame of part i
$\|v_x\|$	Magnitude of velocity of X direction
$\|v_y\|$	Magnitude of velocity of Y direction
$\|v\|$	Magnitude of velocity
$\|a\|$	Magnitude of acceleration
t_j	Temporal features of activity j
THLD	Aberration for threshold

CHAPTER 1: INTRODUCTION

1.1 Problem statement

Videos of on-site operations contain a wealth of information. To assess the efficiency of on-site operations, field engineers usually analyze these videos manually. In the past few years, several researches have made promising attempts to automate assessments using computer vision algorithms. While progress has been made in workers and equipment tracking, progress in activity recognition has been slow. This is primarily because activity recognition requires higher-level reasoning, which is more challenging for computers.

1.2 Scope of the research

In this thesis, we specifically focus on activity analysis for equipment. We developed a novel computer vision algorithm that significantly improves activity recognition accuracy, radically reduces the need for human intervention, and can process a video stream in near real-time. These contributions have important implications in practice: less labor time, less computation (appealing to mobile devices), and real-time activity analysis. To do so, our end-to-end method inputs a video sequence, involves part detection, part tracking and reasoning about the activity, and identifies equipment activity at each frame. Figure 1.1 shows the overview of input/output and our process steps. In the next few sections, we discuss

related work, the algorithm design, and the experimental results.

1.3. Overview

The Figure 1.1 shows the pipeline of our video-based action analysis method. It is mainly composed of user annotation, observation (detection & tracking), and action recognition. Feature selection and extraction is discussed in section 3.1. Detection is discussed in detail and we proposed both HOG + SVM detection pipeline and novel detection method of using CNN in section 3.2, 3.4 and 3.8. We use Kalman filter in tracking stage (see section 3.5 for parameter configuration). In section 3.6, the most important part of activity analysis is discussed.

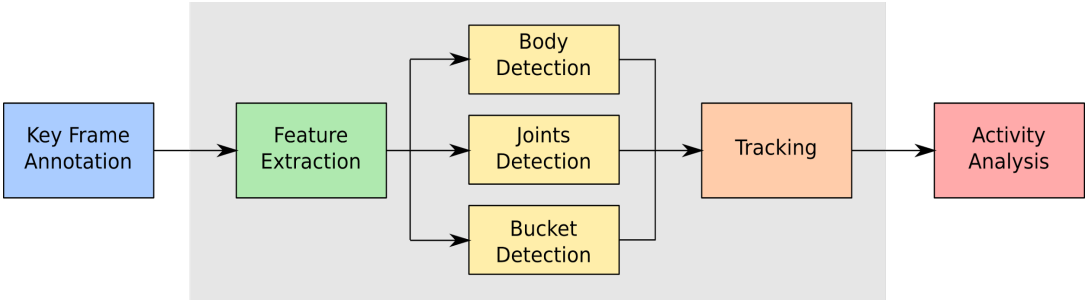


Figure 1.1: Pipeline of the proposed video-based activity analysis method

CHAPTER 2: RELATED WORK

The need for continuous benchmarking and improving the amount of time that equipment and craft workers spend on actual construction from one side, and the rate at which site videos are being generated, have accelerated the demand for machine understanding to enable better activity analysis capabilities. An automated activity analysis allows project management to spend less time on assessing the workface, rather spending their time on the more important process of continuous improvement (Teizer 2009, Gong 2011, Golparvar et al. 2013, Khosrowpour et. al. 2014).

2.1. End-to-end solution for activity recognition

Currently, most of construction equipment performance is still analyzed using traditional data collection including direct manual observation (Oglesby et al. 1989), which are labor-intensive (Gong and Caldas 2011, Goodrum et al. 2010) and can be subjective (Golparvar et al. 2011). Techniques to automate the process of 3D localization and tracking of construction equipment using sensor have been explored in recent research studies (Gong and Caldas 2011) to improve the efficiency and safety of operation, in turn, minimize idle times.

A computer vision based solution for construction activity analysis using video cameras typically involves three main steps: 1) detecting construction resources –equipment and workers– from videos; 2) tracking their location in 2D and/or 3D;

and 3) recognizing the time-series of their activities. Over the past few years, many research efforts (Yang et al. 2010, Park and Brilakis 2011-2012, Azar et al. 2011, 2015, Golparvar et al. 2013-2015) have focused on the task of detecting and tracking construction resources in 2D frames and/or in 3D. A detailed review of these techniques can be found in (Yang et al. 2015). Very promising results have also been reported on the task of detection and tracking, yet there has been little attention to the task of activity recognition.

2.2. Detection

Recent method of using Convolutional Neural Network to estimate human pose has been proposed (Wang et al. 2011). The method is formulated as CNN regressors, which results in high precision pose estimation and has the advantage of reasoning about pose in a holistic fashion. The pose estimation is formulated towards human body joints, which can be easily applied to construction equipment such as excavator that has body-like features such as arm and joints. Other detection technique such as using Histogram of Oriented Gradient (HOG) as the feature descriptor and then training the detector by Support Vector Machine has been widely adopted (Dalal and Triggs 2005, Suykens and Vandewalle 1999). And recently, HOG + Color is introduced and largely improves the detection accuracy by adding pixel color into feature space (Memarzadeh and Golparvar-Fard 2013).

2.3. Tracking

In computer vision, the Kanade-Lucas-Tomasi (Lucas and Kanade 1981, Tomasi and Kanade 1991) feature tracker is an approach to feature extraction. It is proposed mainly for the purpose of dealing with the problem that traditional image registration techniques are generally costly. Other tracking techniques such as Kalman filtering, is a state estimation method based on Gaussian distribution (Kalman 1960).

2.4. Action Recognition

As a step toward addressing the problem of identifying sequences of construction activities from a video, several recent methods have focused on inferring construction activities using location information (e.g. Cheng et al. 2013, Rezazadeh et al. 2012, and Yang et al. 2011). Using prior knowledge about activity locations on the jobsite, and/or by combining accelerometers (Ahn et al. 2013), these method infer the state of the resource activities (e.g. idle vs. non-idle). Still distinguishing between two activities that many have the same location, for example "Digging" versus "Swinging" purely based on location information could be challenging.

Others such as Gong and Caldas (2013) and Golparvar-Fard et al. (2013) leverage video sequences captured from construction sites and present computer vision methods for classifying atomic construction activities. By recognizing atomic construction activities –e.g. Digging, Swinging, Dumping, Moving, and Idling for an excavator– we mean classifying the activities of a single resource (e.g. an equipment)

from video sequences wherein each activity is self-contained within a video. In other words, the video starts with one activity of a single resource and ends with the same activity. These activity recognition methods are binary classification algorithms, and thus for long video sequences where in each resource is engaged in several activities are unable to detect whether that activity has just begun and is getting stronger, is at its peak or is returning to its neutral state. Hence, these methods can not be directly applied to construction site videos wherein the starting and ending temporal points of a resource activity and the activity duration are unknown.

The problem of recognizing activities over long sequences of videos are more fundamental and thus in recent years has received attention from the computer vision community. In several studies, the tasks on activity recognition are either already temporally localized (Liu et al. 2009, Niebles et al. 2010), or detecting tasks mainly focus on localizing well-defined atomic activities (Ke et al. 2007). The application of Hidden Markov Models (HMM) for characterizing sequence of activities is also considered by the authors, though such methods still requires accurate detection and tracking and has a computation time larger than 1-to-1 ratio for each video length. Our work is different from previous methods in that we do not make any assumptions on the expected location of the resources, their relationship to construction activities, or even temporally localizing atomic activities. In addition to discovering discriminative segments of video, which represent atomic construction activities, we also model and learn the durations and the transitions between these segments.

Many of the early works in introducing latent variables into discriminative models were motivated by computer vision application, where it is natural to use latent variables to model human body parts or parts of object in detection task. In the computer vision community there are recent works on training Hidden Conditional Random Fields using max-margin criterion (Felzenszwalb et al., 2008; Wang & Mori, 2008).

CHAPTER 3: METHOD OVERVIEW

Our algorithm inputs a video sequence and identifies equipment activity at each frame with a computational time less than 1-to-1 ratios for each video length. We use taxonomy of a few atomic activities (Figure 3.1) for each equipment. For example, activities of an excavator include: Digging, Swinging (loaded and unloaded), Dumping, Moving and Idling. An earthmoving operation can be expressed in terms of a series of these atomic activities. Our algorithm takes the following steps to recognize equipment activity at each frame:

1- Extract visual features at each frame.

2- Detect and localize characteristic *parts* from the equipment.

3- Track the equipment and its parts and incorporate smoothness criteria.

4- Extract kinematic information such as displacement, velocity, acceleration, and relative positioning.

5- Classify activities by traversing the decision tree (same as taxonomy tree).

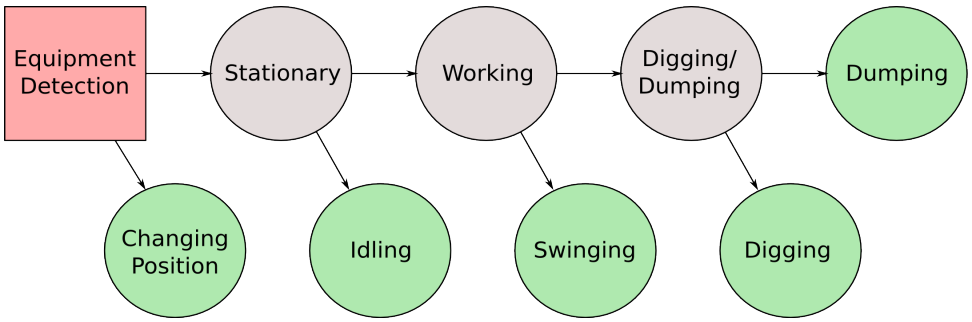


Figure 3.1: Taxonomy tree of the activity analysis. The red box represents the root node. Gray nodes are intermediate states. Green nodes are leaf nodes

3.1. Part Selection

In order to analyze activities, we detect and track a few parts (Figure 3.2, right). Certain parts provide more reliable signal than others. Therefore, we examined several configurations of parts and identified a set of parts that maximize activity recognition accuracy. Besides the body and the bucket of the excavator, we examine four mechanical joints as part candidates. These parts include: the joint between body and arm, the joint between arm and bucket, and the two joints on the excavator's arm. We measure the performance of all part candidates in our activity recognition model. For an excavator, our results show that the best combination of parts includes: excavator body, the top joint, and the bucket (Figure 3.2). Adding more parts does not significantly improve activity recognition accuracy.

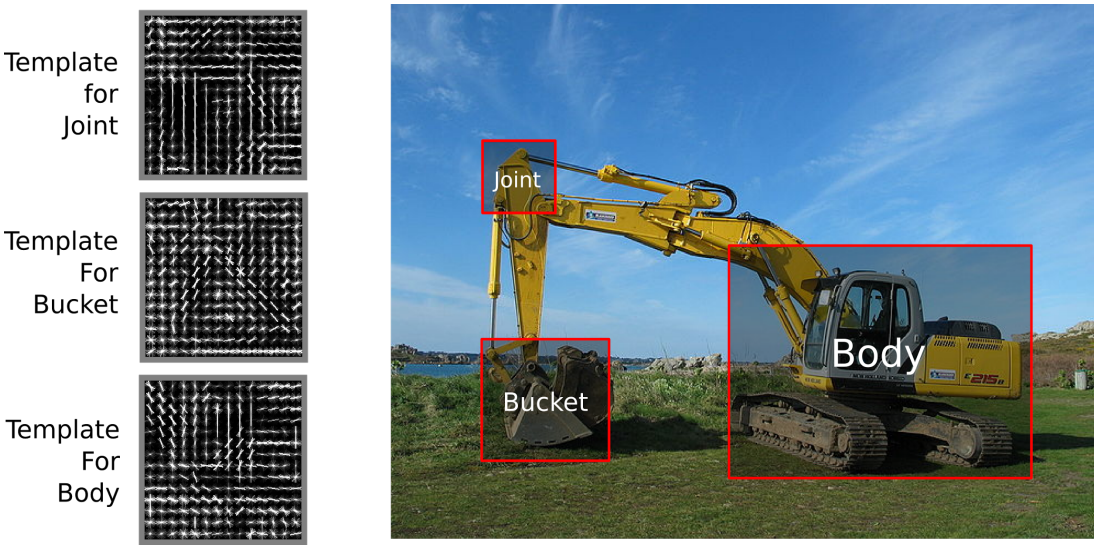


Figure 3.2: Right: The three parts of the excavators to analyze the kinematic characteristics of equipment. Left: Sample HOG templates to detect each part

3.2. Part Detection

We detect parts using Histogram of Oriented Gradients (HOG) features and templates (Figure 3.2, left). We construct training examples using HOG (Dalal and Triggs 2005) and train templates using Support Vector Machine (SVM) (Suykens and Vandewalle 1999). To detect parts on a single frame, we run the part template over the image using sliding window searching method. To process a video, we run templates on all frames of the video sequence. After a detection round, we track parts using a Kalman filter tracker (Kalman 1960, Julier and Uhlmann 1997).

3.3. Training Dataset

Training data includes a set of images with bounding-boxes for each part (Figure 3.2, right). To cover different poses and viewing angles, each part is represented using one out of 3 to 4 templates. Each template is trained with about 500 examples, the total training dataset consists of about 10,000 examples. During detection and tracking, one out of every k frames is considered a key frame. A user can assist tracker by annotating these key frames. We have a user interface that allows the user to pinpoint part locations on each key frames.

3.4. Fast Detection using HOG

We extract HOG features from input images. HOG features are shown to be helpful for object detection and localization (Dalal and Triggs 2005). HOG features

are used to train part templates. In order to detect parts in a new image, part templates are convolved with the corresponding HOG features. In order to cover a range of scales, we use a spatial pyramid of HOG features.

To speed up detection process we benefit from *local search*; we limit feature extraction and convolution to a small window. For every frame, we use the track from the previous frame to identify the location and the size of the search window. This technique is very effective for speed up as it limits computation to a very small fraction of the image. We train multiple templates for each part. Different templates for a part, cover different poses and viewing angles. We run all templates within search window and find maximum scores. Finally, we apply non-maximal suppression and choose the highest scoring location and template.

3.5. Tracking

Typically, an on-site video stream is recorded at the rate of 30 frames per second (fps). This frame rate is usually more than enough for large equipments because they move slowly. To speed up tracking, we perform detection once every *t* frames. We use Kalman filter to track part detections. Kalman filter also smooths out part localization and covers missing frames. Figure 3.4 shows the pipeline of detection and tracking in this thesis

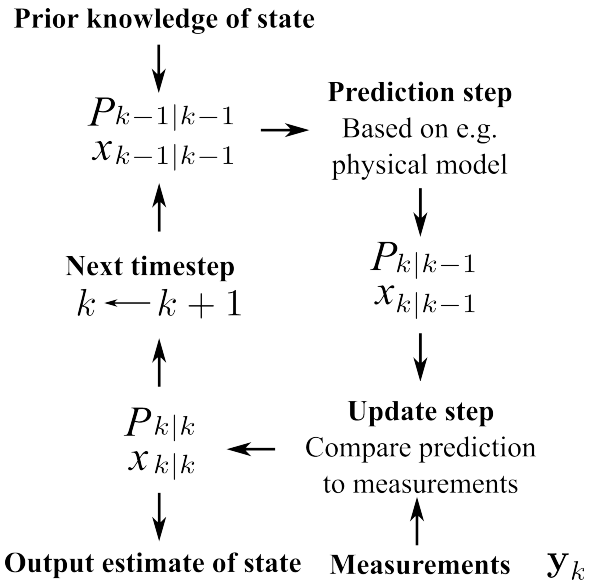


Figure 3.3: Kalman filtering method

3.6. Activity Recognition

To recognize the activity of an excavator we use the taxonomy tree shown in Figure 3.1. This tree works as a decision tree for detection. Leaf nodes in the tree (green nodes) correspond to final states. Parent nodes (red and gray) are intermediate states. Each intermediate state has a binary classifier that classifies the action into two possible states. In order to detect an excavator's activity, we start from the root state (red node) and then traverse nodes until we arrive at a leaf node. This taxonomy tree is designed to maximize accuracy and running time, as it starts with high accuracy classifiers. Each classifier uses various features such as the location (x,y) and the speed (v_x,v_y) of one part or a pair of parts (x_2-x_1,y_2-y_2) (Figure 3.5, 3.7). Root node decides whether the equipment is moving or not. This decision function uses the body's velocity (v_x,v_y) as the main clue. If the velocity is smaller than a threshold,

then the equipment is not moving. The second node, classifies whether the equipment is idle or it is working. This node uses bucket and joint speed as the main clue. If all parts are stationary for a period of time, the algorithm reasons the equipment is idle.

The excavator's arm swings between digging and dumping. The acceleration of the arm is a strong clue to recognize swinging. Our experiments show that a positive and its immediate negative peak in acceleration mark the beginning and the end of a swinging activity (Figure 3.6). We use four constraints to train a classifier for digging and dumping: (1) Whether bucket moves toward body; (2) Digging takes longer than dumping; (3) Dumping is often performed at a higher elevation than digging; and (4) Two diggings in a row or two dumping in a row are unlikely. They often alternate. Our algorithm uses a combination of these constraints to classify digging versus dumping. It should be noted that this taxonomy tree and most nodes are generic and can be applied to equipment other than excavators.

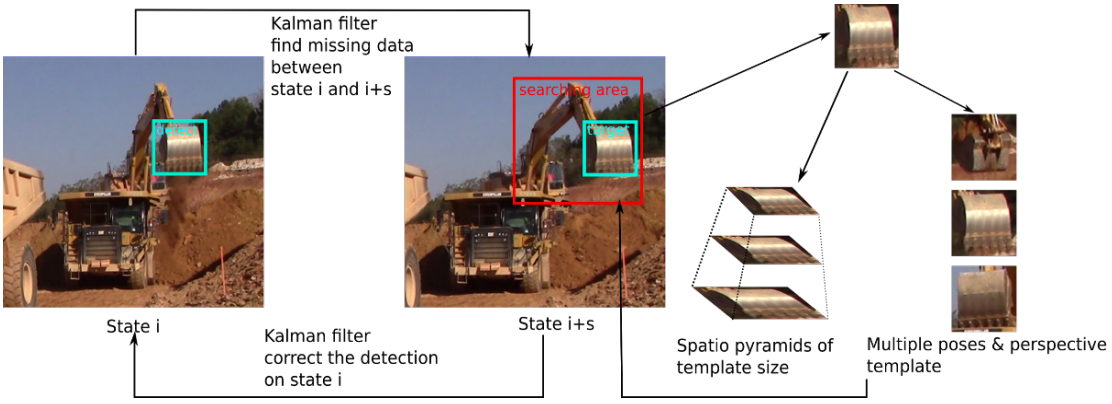


Figure 3.4: This figure illustrate the pipeline of detection and tracking we use in this thesis

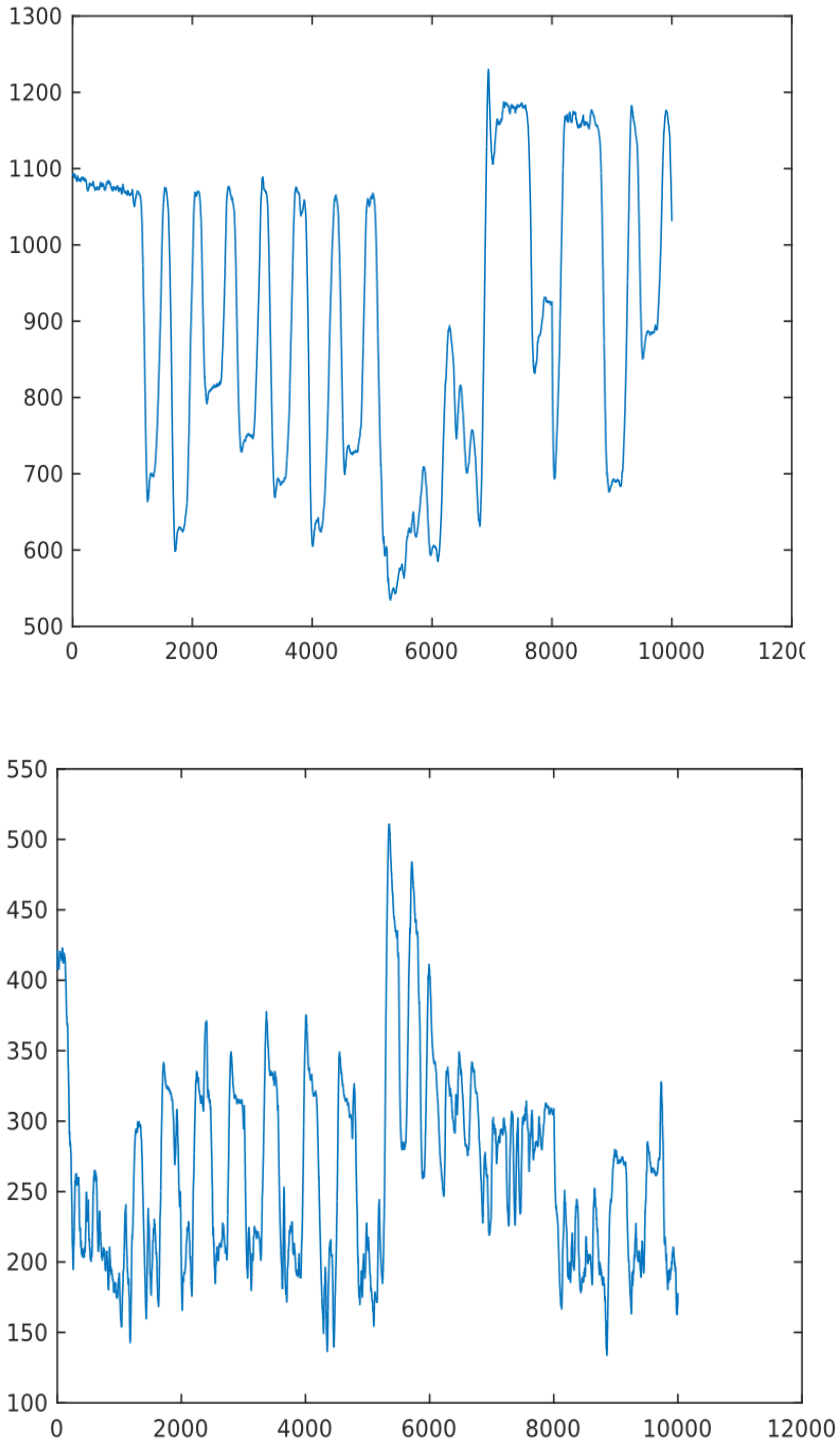


Figure 3.5: Examples of position of parts, which is directly obtained from observation

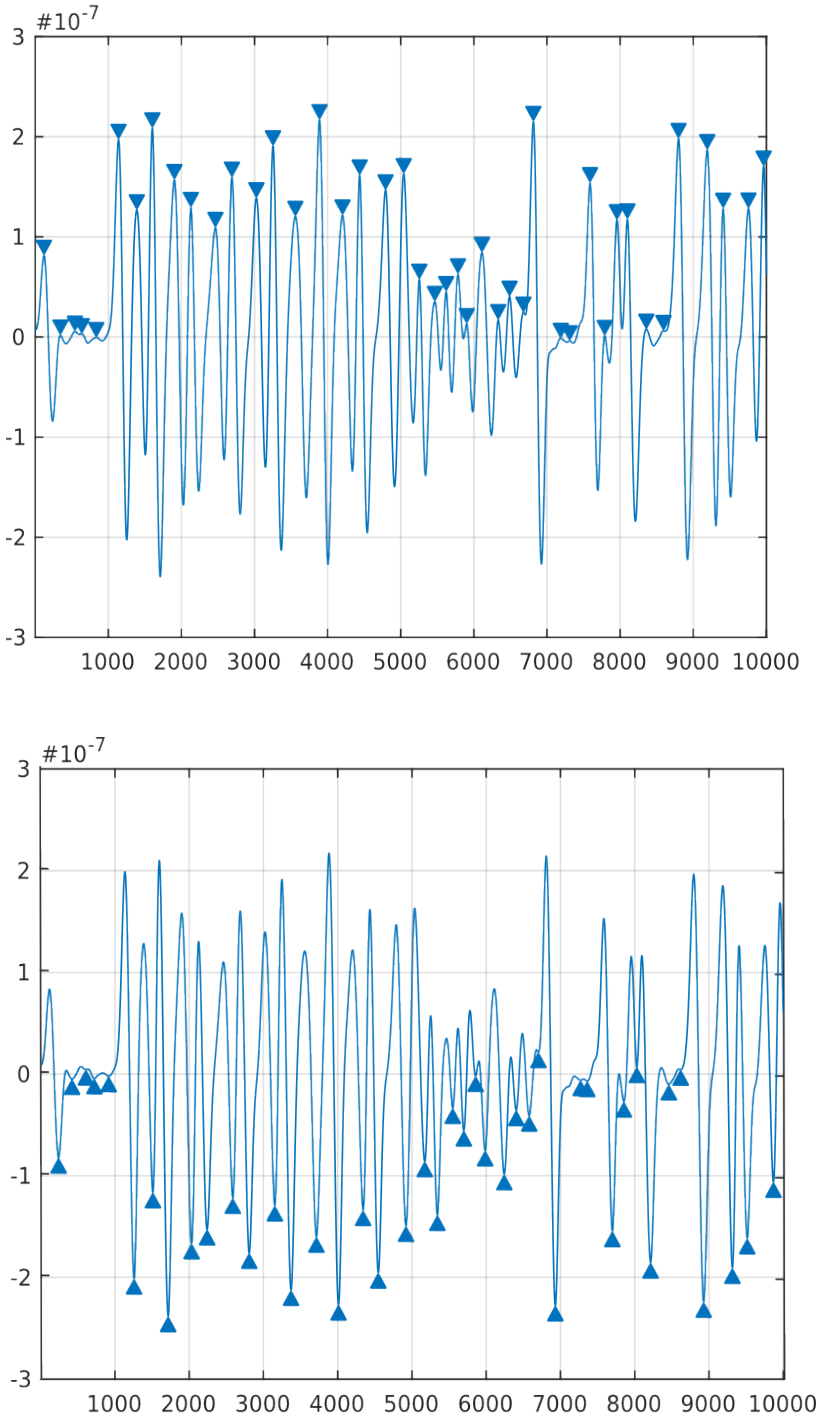


Figure 3.6: Examples of acceleration of parts. Positive and negative peaks identify a kinematic feature to differentiate between swinging and not swinging

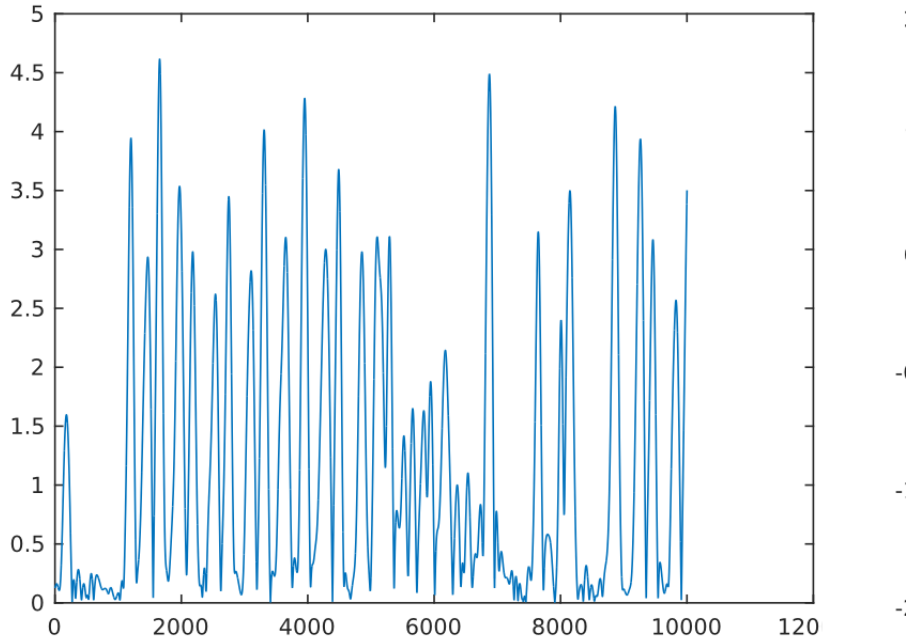


Figure 3.7: Examples of magnitude of velocity of parts, which identify kinematic features to differentiate moving or not as well as digging and dumping. Also it serves as a base for derivation of acceleration

3.7. Latent variable and hierarchical SVM

In statistics, latent variables, (as opposed to observable variable), are variables that are not directly observed but are rather inferred (through a mathematical model) from other variables that are observed (directly measured). Mathematical models that aim to explain observed variables in terms of latent variables are called latent variable models.

In our case, the kinematic features derived from originally observed (x_1, y_1), (x_2, y_2), (x_3, y_3), namely, v_{xi}, v_{yi}, v, a, $\pm Peak_a$ (positive/negative peaks of acceleration time sequence) is so-called latent variable. So we here name them as kinematic latent

variables and their latent variable model is as following:

$$v_{xi} = (x_{i1} - x_{i2})/\Delta t \tag{3.1}$$

$$v_{yi} = (y_{i1} - y_{i2})/\Delta t \tag{3.2}$$

$$v = \sqrt{v_x^2 - v_y^2} \tag{3.3}$$

$$a = \Delta v / \Delta t \tag{3.4}$$

$$\pm Peak_a = peak\ (a) \tag{3.5}$$

We also have three temporal observable variable t_1, t_2, t_3. t_1 is the duration for a atomic "moving" activity, where the duration v_3 lower than the THLD should > t_1; t_2 is the duration for atomic "idling" activity, where the duration v_1,v_2 lower than the THLD should > t_2; t_3 is the duration for atomic "digging/dumping" activities, where t_3 itself is a THLD to differentiate between digging and dumping. Figure 3.8 shows the action recognition model configuration and how to use the kinematic latent variables to recognize equipment activity.

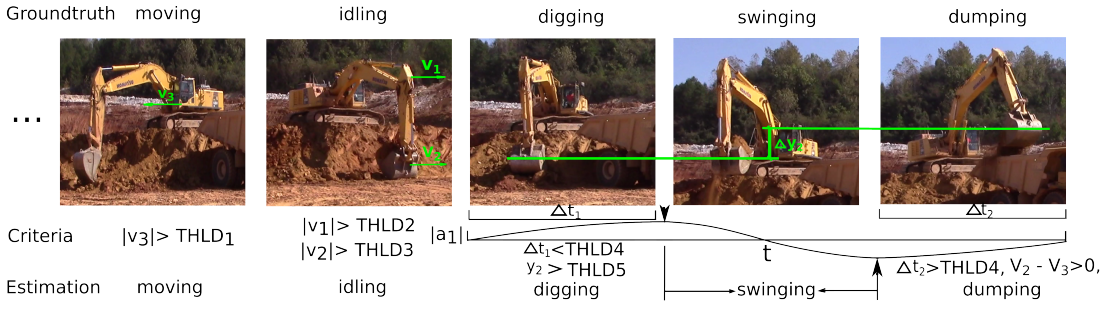


Figure 3.8: Illustration of kinematic and temporal feature based action recognition model

Support vector machines are supervised learning models with associated

learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. Given a set of training examples, each marked for belonging to one of two categories. An SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier. Formula (3.6) shows the basic mathematical model of SVM, in which, y_I is the category of SVM classification, x_i is the feature space of SVM classification, b_i is a constant, and w is weight.

$$y_i \cdot (w \cdot x_i - b) \geq 1 \tag{3.6}$$

In our case, according to our pre-defined action recognition model, there are 5 thresholds need to be determined. SVM provides us with a perfect supplement to our heuristic model, helping training these thresholds using manual annotated data. Figure 3.9 shows a four-level of classification using SVM and heuristic model based on following taxonomy structure.

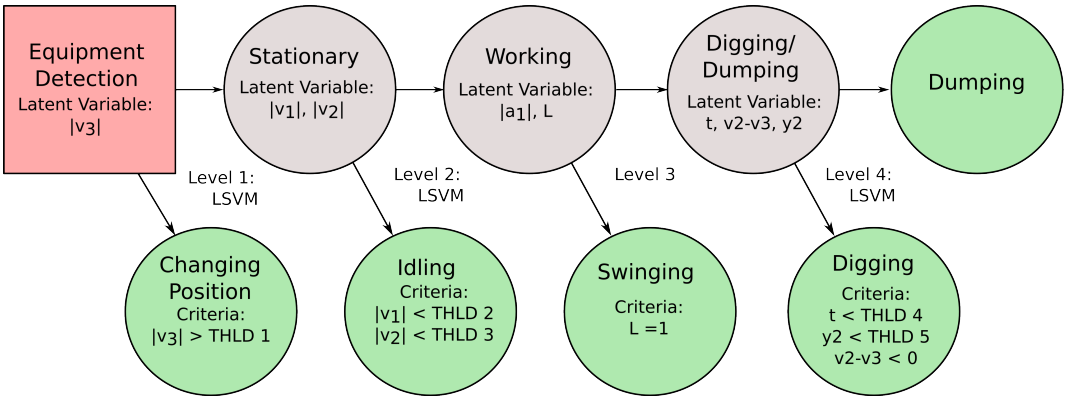


Figure 3.9: taxonomy structural classification for action recognition and parameter

3.8. Alternative detection method -- CNN

For detection, Convolutional Neural Network could be an ideal alternative for HOG feature template plus LSVM to detecting and localizing parts. Because it has following advantages:

1. Detection rates will be reduced by order of magnitude;
2. Directly pinpoint the key parts location instead of drawing a bounding box surrounding them, which avoids unnecessary computing complexity.

Convolutional Neural Network Convolutional Neural Network (CNN) is an extension of Neural Network (NN). NN is a kind of simulation of human brains. A simple NN often has 3 layers: an input layer, a hidden layer, and an output layer, as shown in Figure 3.10. There are weighted connections between elements of adjacent layers (but not within layers). A threshold function is applied on the hidden layer in order to simulate non-linear relations between outputs and inputs. In this case, after training the NN with a labeled dataset, the NN can predict non-linear relations (the same type of relations with training dataset and its labels) from the inputs and generates outputs (Basheer and Hajmeer 2000, Chi and Caldas 2011).

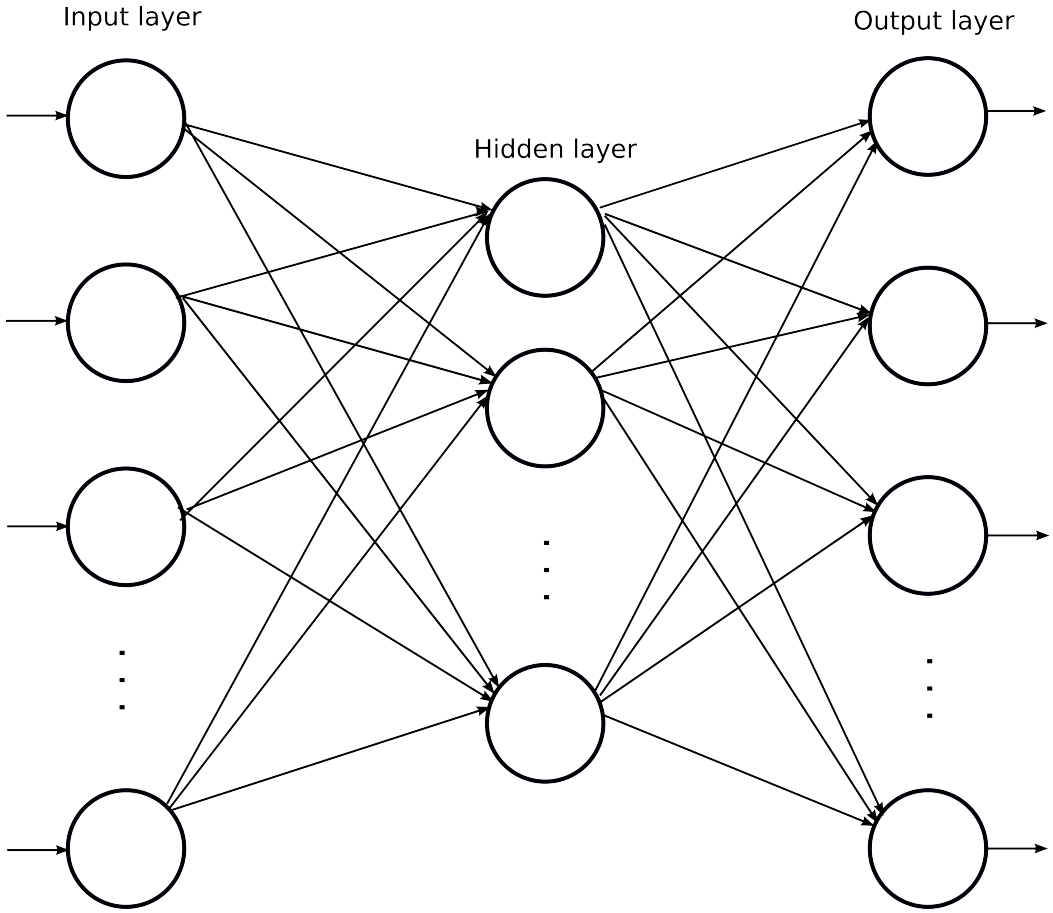


Figure 3.10: Neural Network illustration

CNN is an extension of NN which focuses mainly on 2D data like images [3][10]. Instead of using full connections between layers, CNN uses image filtering (convolution), which connects each element with only a local area of the previous layer. In this way, CNN considers more of 2D structures (like texture) in the data, compared to full connections. So CNN is very suitable to process images. Also, image filtering uses much less weights than full connections, which saves memory and storage. As shown in Figure 3.11, a simple Convolutional Neural Network consists of 2 convolutional layers ($L_C = 2$), 2 pooling layers ($L_P = 2$), and a full-connection layer. In convolution layer, the image convolutes with a set of filters. Each filter convolutes

with the image (across channels) and outputs a feature map. After training, each filter represents a feature in the images, and the feature maps show locations and strengths of each feature. In pooling layers, the feature maps are divided into square blocks (within channels), and a function will be applied on all values in a block to get one output value. A widely used pooling is max-pooling, which chooses the largest value in the block as the output. Unlike image filtering (convolution), the pooling blocks have no or little overlapping, so the output feature maps are much smaller than input ones. Pooling makes feature maps invariant to slight translation and rotation (Bouvrie 2006, LeCun and Bengio 1995).

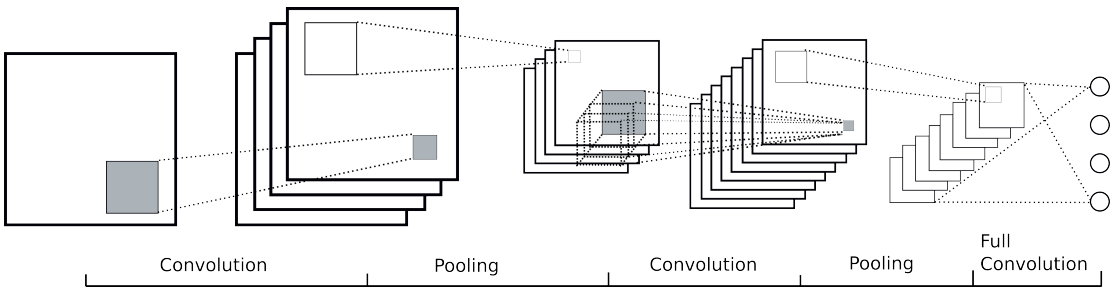


Figure 3.11: Our design of Convolutional Neural Network KLT tracker

CHAPTER 4: EXPERIMENT METHOD

4.1. Detection

4.1.1. HOG + SVM

For evaluating the detection accuracy, a 5 folds cross validation model is applied. Namely, we randomly pick 20% data from both positive and negative dataset to train the classifiers for 5 times. The remaining 80% data of each fold are used for binary testing of classification.

4.1.2. Convolutional Neural Network

In this thesis, we refer to recently proposed method of pose estimation using convolutional neural network (CNNs). The CNNs model has been developed to detect key parts location. And qualitative evaluation has been also conducted and represented in section A.4. We suggest that CNNs will perform faster and better results in detection and localization. The experiment of using CNN detection pipeline includes feature selection, training, testing and evaluation.

Feature Selection:

The feature we select to represent the excavator poses is the five mechanical joints of excavator arm. We name this five joints as truck-arm connection, middle joints, top joints, arm-bucket connection, and bucket end, as show in Figure 4.1.

Figure 4.1: Feature for training the CNN detection. We choose 5 important mechanical joints as the features to be fed into neural network model

Training:

We used a deep learning framework called Caffe (Jia et al. 2014). It provides all the codes of training algorithms, all we need to do is to write network structures. For training data, we used images captured from some videos of construction equipment. We picked out a box containing the construction equipment in each image and resize it to a specific size so that we can feed it into the network. Here we used 2 sizes: 227x227 and 64x64, each with 3 channels of RGB. Then we normalized the values of pixels to RGB values of pixel 128 ! 1 so that the input values are in [-1, 1].The label is the x and y coordinates of the 5 joints of the construction equipment on the image, also normalized into [-1, 1]. We totally have 19080 images for training. The first convolution layer of our network has 16 filters of size 9x9. The first pooling layer performs 3x3 max-pooling with 2 pixels of stride (overlapping 3 pixels between adjacent blocks). The second convolutional layer has 128 filters of 7x7. The second pooling layer is the same as the first one. After pooling of each pooling layer, an activation function f(x) = max(0, x) is performed on each element in order to pick out strong features. After the activation function of the second pooling layers, all the

elements are connected with the 10 elements of the output layer. The connections are weighted full connection, and the 10 elements are our outputs. With the label of each image and the actual output, the program train the network to make the outputs get as close with the labels as possible. The training is processed on batches of images. In our network, each batch contains 8 images, each iteration process one batch of images. We train the network for totally 300,000 iterations, which is using each image for around 126 times. The base learning rate is 0.001, and is decreasing following the law $lr = 0.001(1+ 0.0001 \times \text{iteration})!0.75$ in order the loss function of the network converges well.

4.2. Localization

We use sliding window search method to localize the parts. In detail, each frame is cut into small windows and each window has 20 pixels overlap area. Then, we extract HOG features from each piece and feed them into linear support vector machine to get its classification score. We choose the location of the piece with highest score as our predicted location. Next, we calculate the overlap ratio between the predicted location of the parts and the ground truth. If the ratio exceeds 50%, we believe the localization is correct.

4.3. Tracking

We use Kalman filtering to improve the frame-based detection results of three parts of the excavators. The evaluation of improvement is measured by the

localization accuracy before and after Kalman filtering.

4.4. Activity Recognition

We measure the activity recognition results by producing ground truth manually for testing videos. The ground truth is composed of sequent atomic activities lists and the frame number range for each of them (APPENDIX B). e.g.: frame No. 100 ~ 500 is “Digging”. And our model of action recognition produce the estimated activity sequence for same input video and format. We compare ground truth and our estimation datasets frame-by-frame and calculate the overall accuracy for each activity and how likely they are confused as each other. By representing the evaluation both qualitatively and quantitatively, we use activity timeline illustration with video snapshots as well as the confusion matrix method.

CHAPTER 5: RESULTS AND DISCUSSION

5.1. Input data

We produced a dataset to experiment with equipment activity recognition. This dataset includes a combination of excavators and dump trucks for five types of excavator activities (i.e., moving, digging, swing, dumping, idling). Our dataset covers variations of equipment shape, camera viewpoints, lighting conditions, and occlusions. We include Komatsu and Caterpillar excavators and Caterpillar dump trucks. Table 5.1 shows the configuration of the dataset. We run experiments on a machine with a 3.6 GHz Intel Core i7 processor and 64GB of memory.

Table 5.1: Input data configuration

	Camera type	Video Size	Length	# of frames	Equip. type
VIDEO 1	Commodity HD Video Camera	1920 × 1080	20 min	36’000	Excavator & Dump truck
VIDEO 2		1920 × 1080	25 min	45’000	
VIDEO 3		1920 × 1080	33 min	59’000	

5.2. Evaluation for detection

5.2.1. HOG + SVM

We produced a dataset to experiment with equipment parts detection. This dataset includes three parts of the excavator (i.e., the top joints of excavator arm, the bucket, the body). For each part, we prepared 1000 positive data divided to 3~4 templates. Multiple templates can reduce the false detection caused by different pose of the excavator, or different viewer perspectives (Figure 5.1).

Figure 5.1: Templates for three parts of the excavators, we train multiple templates fore each part to reduce the false detection because of different viewer perspective or equipment poses variation

Then we extract HOG template from each of them, where block size is S_B = 2 ×2, and cell size S_C varies from 4×4 to 10×10 depends on both the part and its template we choose. Next, we feed HOG templates into linear support vector machine to train the classifiers for templates of each part.

The average accuracy turns out to be 85.6% using single template for each part, and 90.1% using multiple templates. The introduction of multiple templates lifts the accuracy by 4.5%. (Table 5.2, Figure 5.2)

Table 5.2: Detection accuracy of using single and multiple templates for each part

	Part	VIDEO 1	VIDEO 2	VIDEO 3	Average
Single template	Arm Tip	0.8701	0.9770	0.5914	0.8225
	Bucket	0.7562	0.9800	0.9371	0.8911
	Body	0.8501	0.7143	0.9990	0.8544
Multiple template	Arm Tip	0.8701	0.9920	0.6294	0.8305
	Bucket	0.9880	1.0000	0.9570	0.9817
	Body	0.9001	0.7752	1.0000	0.8917

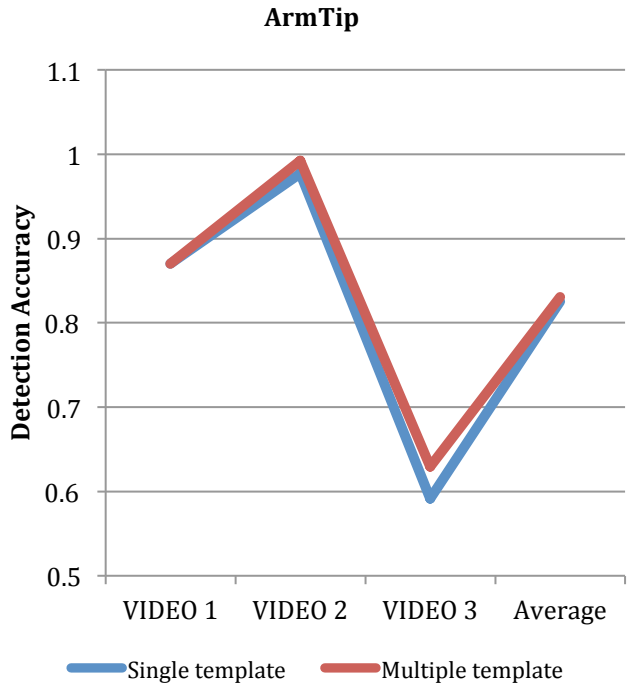
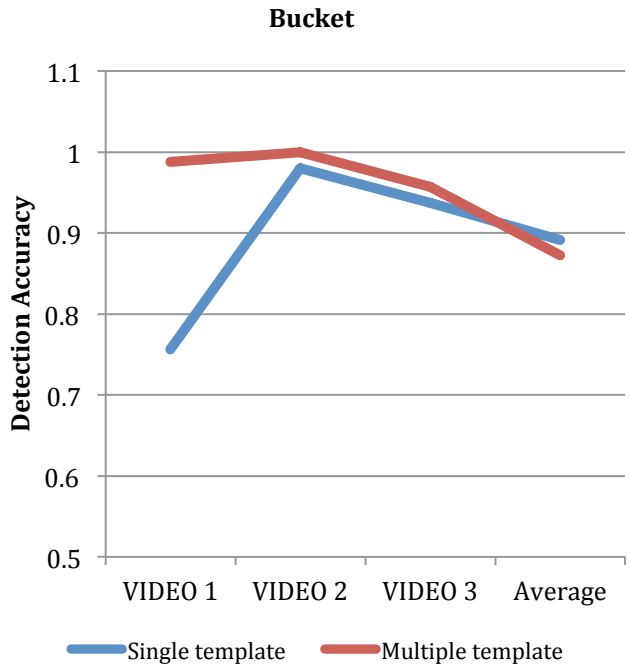


Figure 5.2: Detection accuracy curves of using single and multiple templates for each part



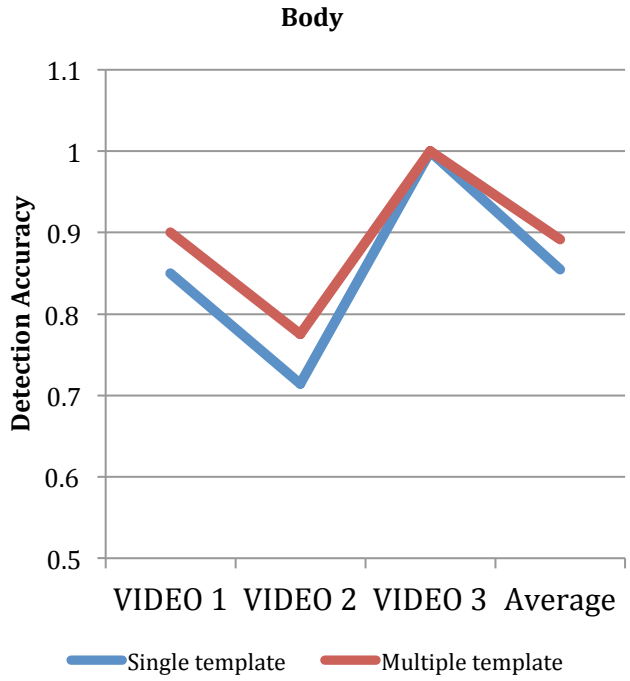


Figure 5.2 (cont.)

5.2.2. Convolution Neural Network

We first use linear SVM to train a HOG-based excavator detector, and then use it to localize the excavator in each testing frame. Next, we convolve the region-containing excavator with the detector we trained to obtain the estimated location of the five key joints representing excavator poses. Testing data size is 100 frames for each video. The comparison between ground truth and estimation is show in Figure 5.3. White point is the ground truth, color strip stands for the estimated poses by connecting key point detected together. The top row shows the best estimation results, the middle row shows the fair results, and the bottom row shows the worst results. The result shows that resizing images to 64×64 leads better results than 227×227 size filter in testing clips from Video 2 and Video 3, but worse in Video 1.

Figure 5.3: Illustration of detection results, the results get better from bottom to top (white dots stands for the ground truth of key point, colored line is our estimation using CNN detection)

The estimated position of five key points in each frame from testing video clip will be put into the activity recognition model based on features geo-location and here, in Figure 5.4 we show some qualitative result of activity recognition and its comparison with ground truth using CNN. To obtain quantitative results of activity recognition, we applied HOG + SVM detection pipeline, which is discussed in section 5.5.

Figure 5.4: Illustration of activity recognition results using CNN detection (The left top three lines: red – the frame ID, green – our estimate activity, yellow – the ground truth)

5.3. Evaluation for localization

We produce a dataset to experiment with parts localization. This dataset includes 3000 video frames with bounding box annotation surrounding each part (Figure 5.5). The average localization accuracy reaches 43%.

Figure 5.5: Parts localization. We implement sliding windows method to localize the interested parts and put bounding boxes surrounding them

The distance between equipment and camera may vary from case to case. Here, the spatial pyramid method is applied to solve this difficulty (Figure 3.3). The testing template we extract from each location will be resized by 0.5-3 times of its original size. And each size for template will be examined and scored by classifiers. Then non-maxima suppression is applied to choose the best-scored template size. Table 5.3 and Figure 5.6 shows the average localization accuracy and its variation of

each part of three testing videos.

Table 5.3: Localization accuracy of three parts of the excavators

	VIDEO 1	VIDEO 2	VIDEO 3	Average
Arm Tip	0.4745	0.3673	0.2816	0.3745
Bucket	0.5398	0.2541	0.4214	0.4051
Body	0.3551	0.8153	0.4000	0.5237

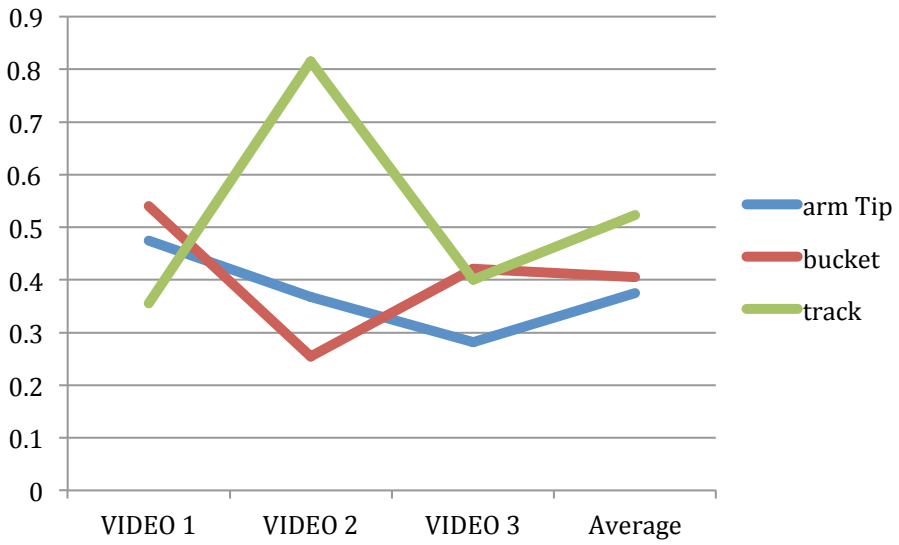


Figure 5.6: Localization accuracy curves

5.4. Evaluation for tracking

We also use Kalman filter tracker to further improve the localization accuracy. The configuration of the Kalman filter is as follow:

State transition model is

$$ST = [1\ 1;\ 0\ 1];$$

Measurement model is

$$M = [1\ 0];$$

Process noise *PN* = 1e-4; Measurement noise *MN* = 4. The following figure shows the comparison of observation before and after Kalman filtering. Figure 5.7 shows the curve of parts' position before and after Kalman filtering. We can see the smoothness and elimination of outliers applying this technique.

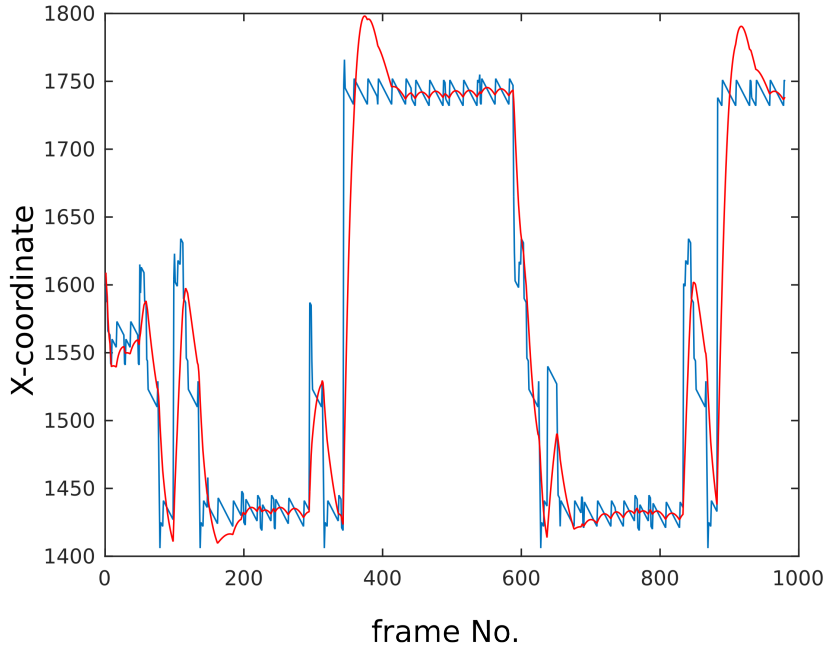


Figure 5.7: Curves of parts position X, Y coordinates. The blue/red lines indicates the X and Y coordinate curve before/after Kalman filtering

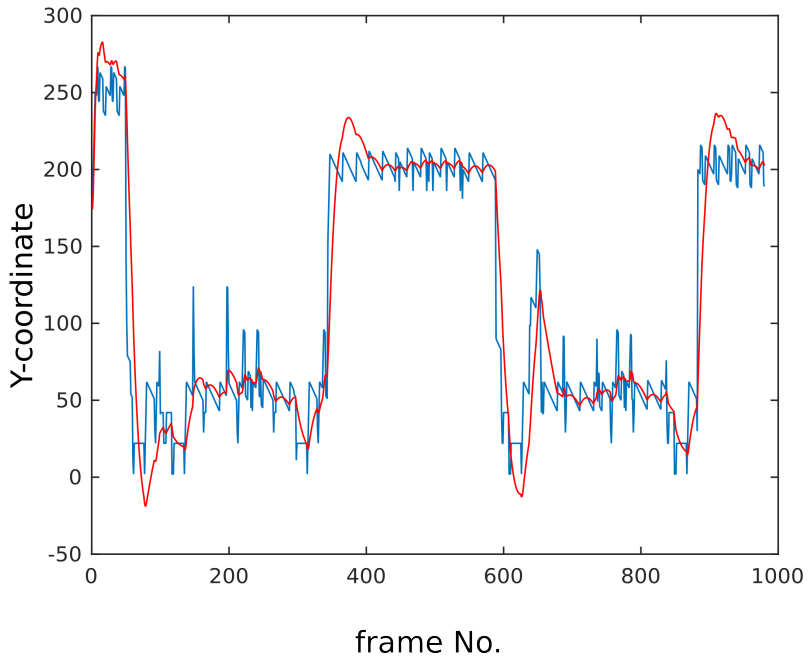


Figure 5.7 (cont.)

From the Table 5.4, we can see that applying Kalman filtering improves the localization accuracy by 4.8%. Figure 5.8 also shows us the comparison of accuracy before and after Kalman filtering.

Table 5.4: Comparison of localization accuracy before and after Kalman filtering

	Arm Tip	**Bucket**	**Track**
Before Kalman filtering	0.3745	0.4051	0.5237
After Kalman filtering	0.4466	0.4446	0.5432

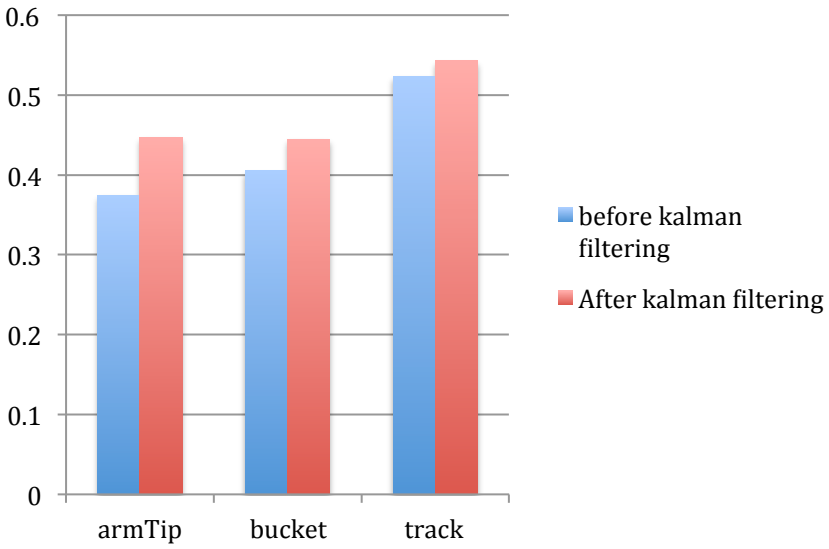


Figure 5.8: Illustration of accuracy improvement for localization after Kalman filtering

5.5. Evaluation for activity recognition

The feature space of activity classification is a thirteen dimensional vector $F(n)$, $n = 13$, in which:

1. Six dimensions are the x, y position of all three parts, x_1, x_2, x_3, y_1, y_2, and y_3;

2. Two dimensions are the magnitude of velocity of part "Arm Tip" and "Body", $|v_1|$, $|v_3|$;

3. One dimension is the orientation of relative velocity between parts "Bucket" and "Arm Tip", v_2-v_3;

4. Three dimension are the temporal feature including the duration of conducting atomic activity of "Moving", "Idling", "Digging/Dumping", t_1, t_2, and t_3;

5. One dimension is the acceleration of part "Arm Tip", a;

So the feature vector is as follow:

$$F(13) = [x_1, y_1, x_2, y_2, x_3, y_3, |v_1|, |v_3|, v_2\text{-}v_3, t_1, t_2, t_3, a] \quad (4.1)$$

We measure performance both qualitatively and quantitatively. Figure 5.9 illustrates three instances of activity recognition each over a period of one minute. This figure compares our activity recognition with the ground truth.

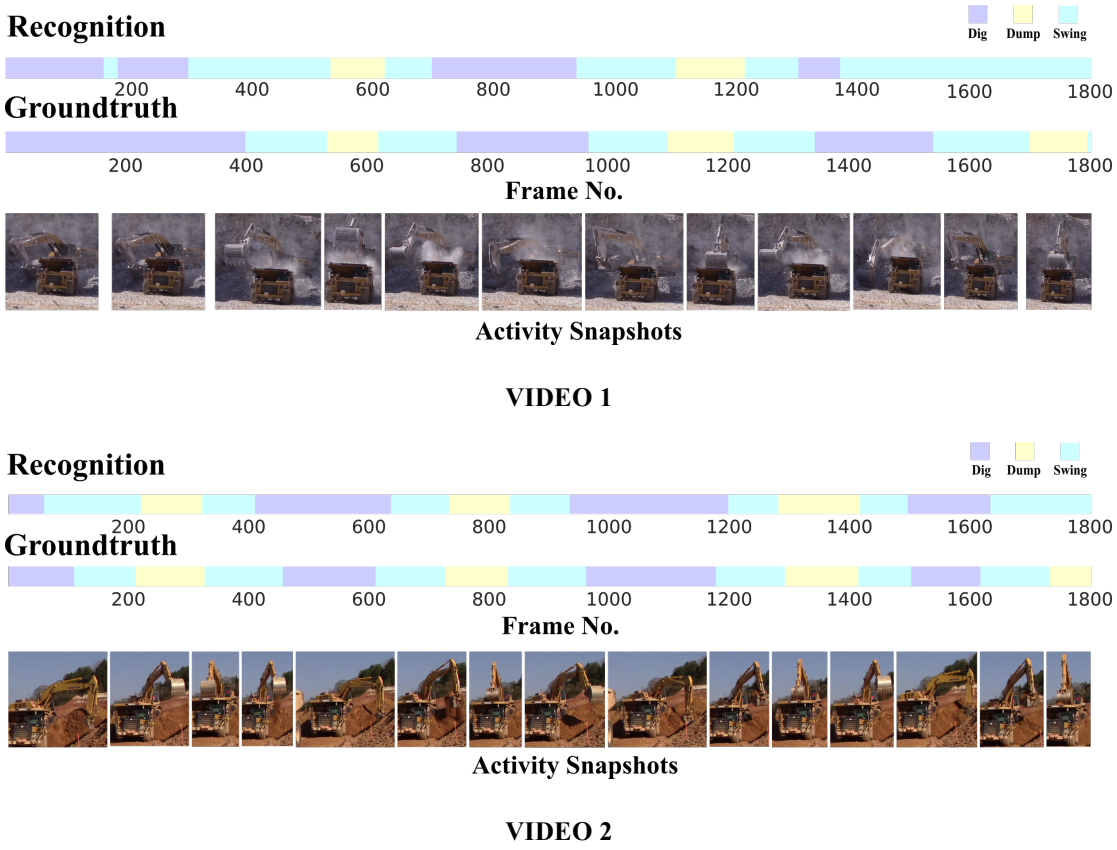


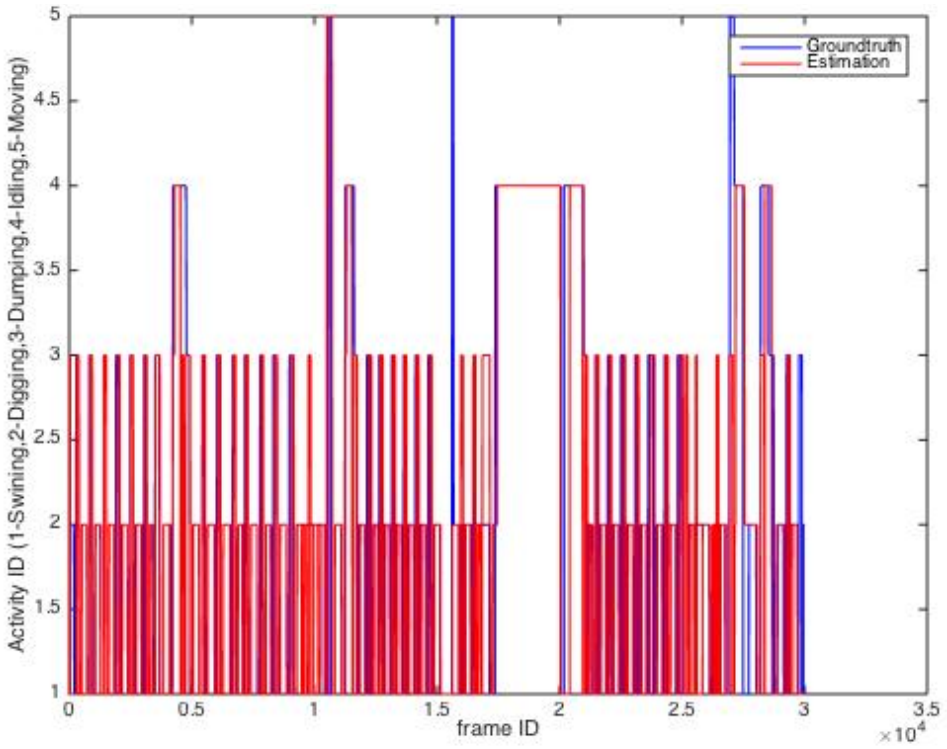
Figure 5.9: The comparison between timeline illustration of our recognition and the ground truth and the activity snapshot corresponding to ground truth for one minute (From top to bottom: VIDEO 1, 2, 3)

VIDEO 3

Figure 5.9 (cont.)

From the upper chart of Figure 5.10 and 5.11, we can see that our estimation matches the ground truth well. Even though sometimes moving will get confused, it is explainable regarding the fact that the feature on excavator identifying moving can be easily blocked by the trucks, but moving usually happens when truck is out of the field of view. This is discussed in the following content. The lower chart of Figure 5.10 and 5.11 perfectly illustrates the kinematic nature of our activity recognition model by showing the variation of parts' x, y coordinates and it's corresponding atomic activity sequence. Taking "idling (black)" as an example, the black area (Figure 5.11 button) matches the little variation of parts' x, y coordination (equal to small magnitude of velocity). This is reasonable, because in our model, we assume that "idling" happens when the magnitude of velocity on both parts "Arm Tip" and "Bucket" is smaller than the threshold.

The variation of coordinates is quite consistent with the activity transition especially for digging, dumping, swinging. This also proves our analysis (see discussion below) that most of error is caused by the different timings between ground truth and detection, rather than false recognition.



Action: pink – digging, yellow – swinging, green – dumping, black – moving, white – idling

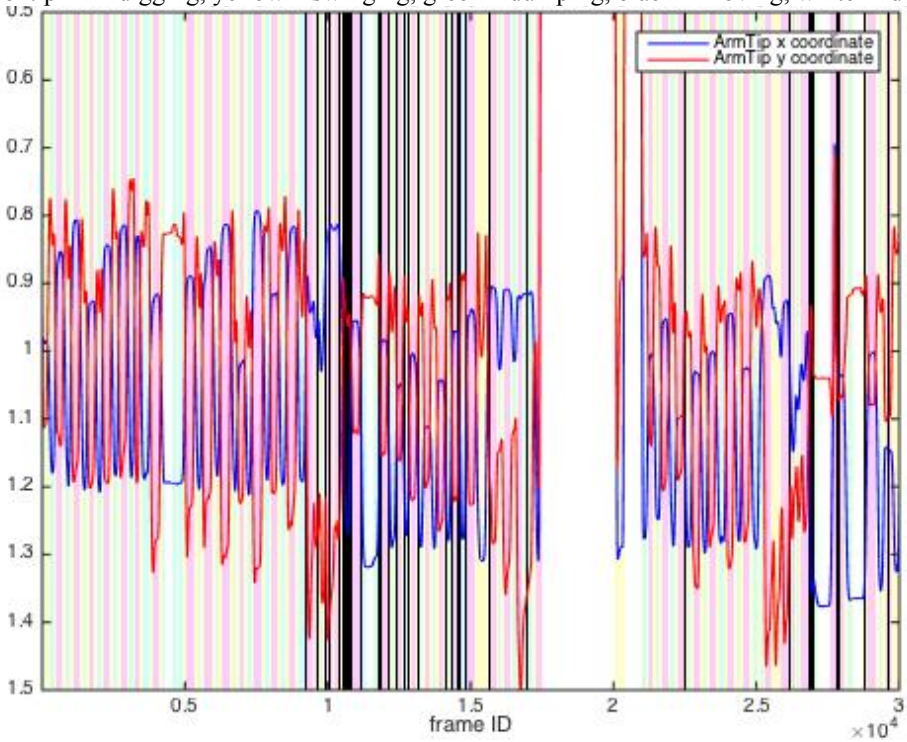
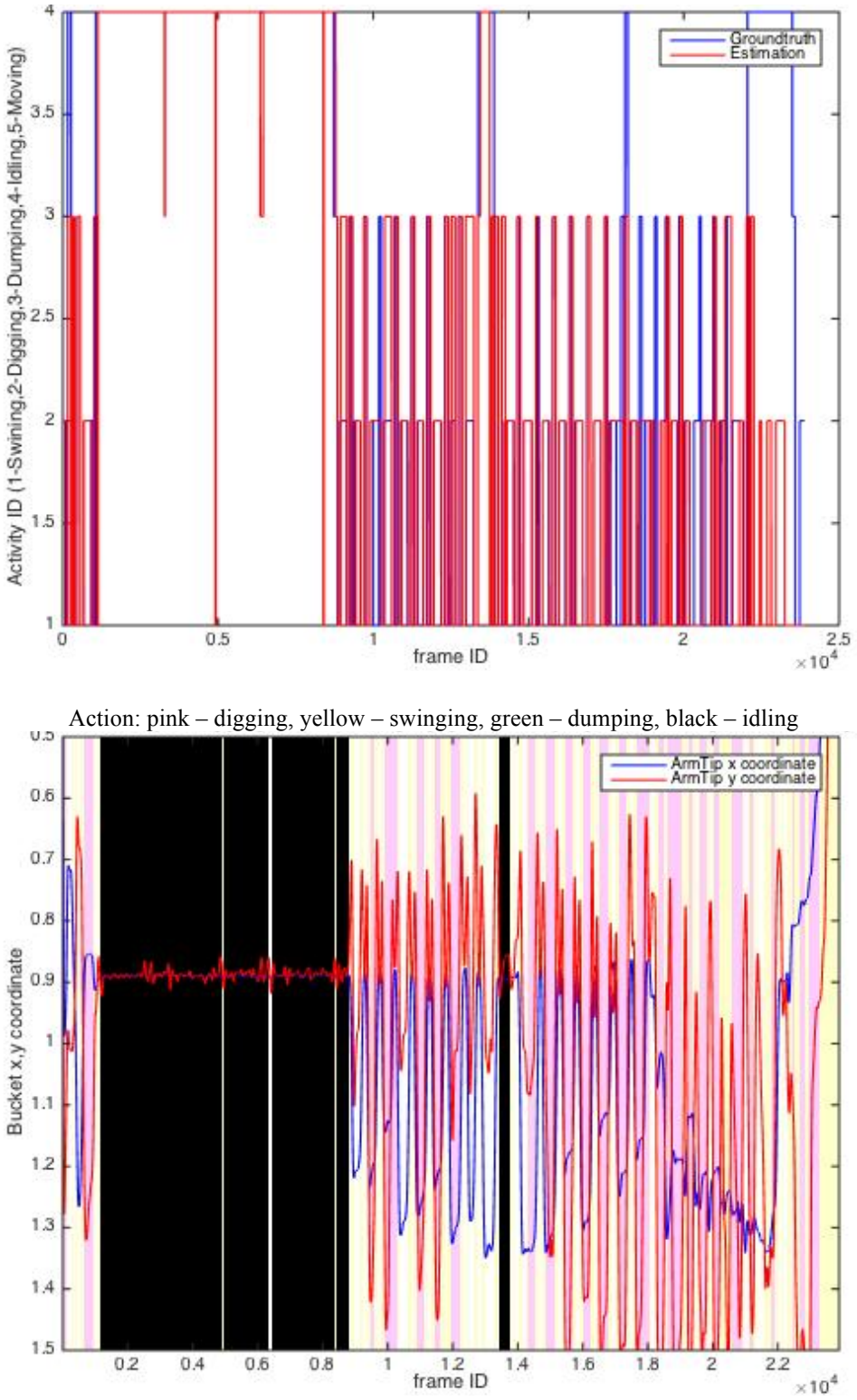


Figure 5.10: High temporal resolution comparison between ground truth and estimation (upper), activity recognition with according x, y coordinates of part (lower) "Arm Tip", VIDEO 1



Action: pink – digging, yellow – swinging, green – dumping, black – idling

Figure 5.11: High temporal resolution comparison between ground truth and estimation (upper), activity recognition with according x, y coordinates of part (lower) "Arm Tip", VIDEO 2

We measure performance using confusion matrix. Confusion matrix visualizes the rate of confusion between every pair of activities (Figure 5.12). Our total frame-by-frame activity recognition accuracy is 88.91%. It should be noted that most of the 11.09% error is caused by different timings between ground truth and detection. In most cases of error, our detection is either a few seconds ahead of the ground truth or behind the ground truth. This mode of error does not affect the count of each action or the total time in an action. This effect can also be seen in Figure 5.9.

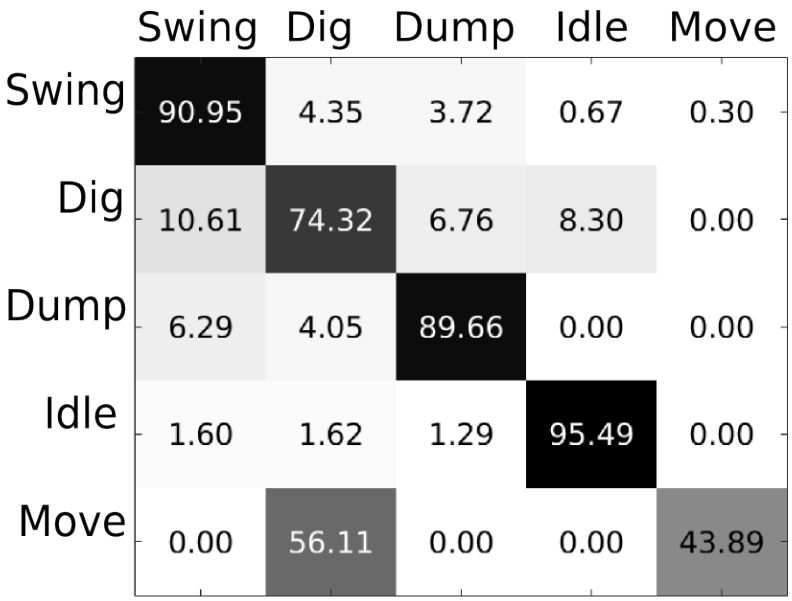


Figure 5.12: Confusion Matrix of excavator activity recognition.(From top to bottom corresponding to VIDEO 1, 2, 3)

Figure 5.12 (cont.)

In Figure 5.12, it is noticeable that “Dumping” and “Digging” are more easily confused with swinging, partially we believe is caused by different turning point selected by machine and human labeler regarding the transition from dumping or

digging to swinging (vice versa).

In the confusion matrix of Video 1, the “Moving” is confused as “Digging” easily. We infer “Moving” from the velocity of part -- “excavator body”, and it might be easily blocked because of the constantly involvement of trucks. We believe this is able to be improved by assuming that the “excavator body” only moves when there are no trucks in the field of view. This assumption depicts the true scenario that in most of the case, excavator moves to change its mining or excavating position only when shifting between trucks happens, namely, no truck in the field of view block our recognition. This can be explained by the intention of construction managers to eliminate equipment idling. Future work will include experiments to optimize this.

There is a trade-off between human-intervention, accuracy and computing complexity. The more user annotation the higher the accuracy, and the lower the computation time will be. Our experiments show that about 1,000 frames between users annotations are optimal in this trade-off (Table 5.5). Further comparison between our method and state-of-the-art method on computing complexity and accuracy will be discussed in APPENDIX A.

Table 5.5: Computing efficiency compared with the previous State-of-the-art method (per frame)

	Label time	**Detection**	**Tracking**	**Recognition**	**Total**
Our Mtd.	0.1~1 min/label	0.16’’	0.03’’	10^-4’’	0.19’’

CHAPTER 6: CONCLUSIONS

We present a new method for automatic activity analysis of construction equipment. Our method has two advantages comparing to the previous methods: It requires much less human intervention and it operates in near real-time. We use a taxonomy of atomic activities including Moving, Idling, Swinging, Digging, and Dumping. Multiple of these atomic activities compose of complex activities. We have a pipeline to detect, track, and recognize activity. Our experimental results with an average accuracy of 89% and maximum accuracy of 98% are promising for automatic activity analysis while being an order of magnitude faster than the previous state-of-the-art. Further researches can be emphasize on improving the performance particularly when the equipment is moving and also validating the application of this framework to other types of equipment.

For CNN detection, even though single point estimation may deviate much from the ground truth occasionally, the connection among 5 key points still accurately represents the right pose of the excavator. Also, the computing efficiency is nearly real-time, the speed reaches least 20 frames per second for testing. So we conclude that this method is suitable for our action recognition model based on the key point position in video frames. This method is suitable for our action recognition model and works more efficient and accurate than using frame-wise HOG/SVM detection with Kalman filtering. Since this method eliminates the problems such as drifts, so we regard it is also more suitable than Lucas-Kanade tracking.

The future work can be focused on: 1) Refining the methodology for action recognition to be less time consuming and fully-automatic. 2) Applying this action recognition system to predict action of other construction equipment; 3) Applying this action recognition system to predict construction workers' activity to avoid safety hazards. For Convolution Neural Network detection, future research efforts can be put on designing and improve network, and enlarge the training dataset to improve the accuracy of estimated feature location.

REFERENCES

C. R. Ahn, P. Lewis, M. Golparvar-Fard, S. Lee (2013), "Integrated framework for estimating, benchmarking, and monitoring pollutant emissions of construction operations", *Journal of Construction Engineering and Management* 139.

E. Rezazadeh Azar, B. McCabe (2012), "Automated visual recognition of dump trucks in construction videos", J. Comput. Civ. Eng. 26 769–781.

E. Rezazadeh, B. McCabe (2012), "Part based model and spatial–temporal reasoning to recognize hydraulic excavators in construction images and videos", Automat. Constr. 24 194–202. I. Brilakis, L. Soibelman (2008), "Shape-based retrieval of construction site photographs", ASCE J. Comput. Civil Eng. 22 (1) 14–20.

E. Rezazadeh Azar, S. Dickinson, B. McCabe (2012), "Server-customer interaction tracker: Computer vision–based system to estimate dirt-loading cycles", J*ournal of Construction Engineering and Management* 139 785–794.

I. Brilakis, M.-W. Park, G. Jog (2011), "Automated vision tracking of project related entities", Adv. Eng. Inform. 25 713–724.

S. Chi, C.H. Caldas (2011), "Automated object identification using optical video cameras on construction sites", Comput.-Aid. Civ. Infrastruct. Eng. 26 368–380.

T. Cheng, J. Teizer, G. C. Migliaccio, U. C. Gatti (2013), "Automated task-level activity analysis through fusion of real time location sensors and worker's thoracic posture data", *Automation in Construction* 29 24–39.

N. Dalal, B. Triggs, (2005, June). "Histograms of oriented gradients for human detection." In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on (Vol. 1, pp. 886-893). IEEE.

M. Golparvar, F. P. Mora, and S. Savarese.(2011) "Integrated Sequential As-Built and As-Planned Representation with Tools in Support of Decision-Making Tasks in the AEC/FM Industry". Journal of Construction Engineering and Management, 137(12):1099–1116.

M. Golparvar-Fard, A. Heydarian, J.C. Niebles (2013), "Vision-based action recognition of earthmoving equipment using spatio-temporal features and support vector machine classifiers", *Adv. Eng. Inform*. 27 652–663.

J. Gong, C. H. Caldas, C. Gordon (2011), "Learning and classifying actions of construction workers and equipment using bag-of-video-feature-words and

bayesian network models", *Advanced Engineering Informatics* 25 (2011) 771–782.

J. Gong and C. H. Caldas (2011). "An object recognition, tracking, and contextual reasoning-based video interpretation method for rapid productivity analysis of construction operations." Automation in Construction, 20(8):1211–1226.

P. Goodrum, M Asce, C. Haas, C. Caldas, D. Zhai, J. Yeiser, and D. Homm (2010). "Model to predict the impact of a technology on construction productivity." Journal of Construction Engineering and Management, (September):678–688.

P. Felzenszwalb, D. McAllester, and D. Ramanan (2008). "A discriminatively trained, multiscale, deformable part model." Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE.

A. Heydarian, M. Golparvar-Fard, J. C. Niebles, "Automated visual recognition of construction equipment actions using spatio-temporal features and multiple binary support vector machines", in: Proc., Construction Research Congress.

Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, and UC Berekely Eec (2014). "Caffe : Convolutional Architecture for Fast Feature Embedding." ACM Conference on Multimedia

S. J. Julier, J. K. Uhlmann,(1997, July). "New extension of the Kalman filter to nonlinear systems." In AeroSense'97 (pp. 182-193). International Society for Optics and Photonics.

R. E. Kalman (1960). "A new approach to linear filtering and prediction problems." Journal of Fluids Engineering 82.1 :35-45.

Y. Ke, R. Sukthankar, "M. Hebert, Event detection in crowded videos", in: IEEE 11th "International Conference on Computer Vision", 2007. *ICCV 2007*., IEEE, pp. 1–8.

A. Khosrowpour, J.C. Niebles, M. Golparvar-Fard (2014), "Vision-based workface assessment using depth images for activity analysis of interior construction operations", Automat. Constr. 48 74–87.

R. Kohavi (1995). "A study of cross-validation and bootstrap for accuracy estimation and model selection." Ijcai. Vol. 14. No. 2.

J. Liu, J. Luo, M. Shah (2009), "Recognizing realistic actions from videos in the wild, in: IEEE Conference on Computer Vision and Pattern Recognition", CVPR 2009, IEEE, pp. 1996–2003.310

B. D. Lucas, and T. Kanade (1981). "An iterative image registration technique with an application to stereo vision." IJCAI. Vol. 81.

J. Malcolm, Y. Rathi, A. Tannenbaum (2007), "Multi-object tracking through clutter using graph-cuts", IEEE International Conference on Computer Vision, p.1–5.

M. Memarzadeh, M. Golparvar-Fard, J.C. Niebles (2013), "Automated 2D detection of construction equipment and workers from site video streams using histograms of oriented gradients and colors", Automat. Constr. 32 24–37.

M. Memarzadeh, A. Heydarian, M. Golparvar-Fard, J. Niebles, "Real-time and automated recognition and 2d tracking of construction workers and equipment from site video streams", in: Int. Workshop on Computing in Civil Engineering.

J. C. Niebles, C.-W. Chen, L. Fei-Fei (2010), "Modeling temporal structure of decomposable motion segments for activity classification", in: *Computer Vision–ECCV, Springer*, 392–405

N. NIST, National Institute of Science and Technology 2011–2012 Criteria for Performance Excellence, 2011.

C. Oglesby, H. Parker, and G. Howell (1989). "*Productivity Improvement in Construction*". McGraw-Hill.

M.-W. Park, I. Brilakis (2012), "Construction worker detection in video frames for initializing vision trackers", Automation in Construction 28 15-25.

M.-W. Park, I. Brilakis (2012), "Enhancement of construction equipment detection in video frames by combining with tracking", in: ASCE International Conference on Computing in Civil Engineering, pp. 421–428.

M.-W. Park, A. Maknmalbaf, I. Brilakis (2010), "Comparative study of vision tracking methods for tracking of construction site resources", Automation in Construction 20 905-915.

P. Sermanet, et al (2013). "Overfeat: Integrated recognition, localization and detection using convolutional networks." arXiv preprint arXiv:1312.6229.

J. A. Suykens, J. Vandewalle,(1999). "Least squares support vector machine classifiers." Neural processing letters, 9(3), 293-300.

J. Teizer, P.A. Vela, "Personnel tracking on construction sites using video cameras", Adv. Eng. Inform. 23 (2009) 452–462.

C. Tomasi, and T. Kanade (1991). "Detection and tracking of point features."

Pittsburgh: School of Computer Science, Carnegie Mellon Univ.

P. Viola, M. Jones, (2001). "Rapid object detection using a boosted cascade of simple features." Proceedings of the 2001 IEEE Computer Society Conference In Computer Vision and Pattern Recognition, I-51.

J. Yang, O. Arif, P. A. Vela, J. Teizer, Z. Shi (2010), "Tracking multiple workers on construction sites using video cameras." Advanced Engineering Informatics 24 428-434.

J. Yang, P. Vela, J. Teizer, Z. Shi (2011), "Vision-based crane tracking for understanding construction activity", Proc. ASCE IWCCE 258–265.

J. Yang, Jun, Man-Woo Park, Patricio A. Vela, and Mani Golparvar-Fard (2015). "Construction performance monitoring via still images, time-lapse photos, and video streams: Now, tomorrow, and the future." *Advanced Engineering Informatics* 29, 2: 211-224.

B. Yao, A. Khosla, and FeiFei Li (2011). "Classifying actions and measuring action similarity by modeling the mutual context of objects and human poses." Proceedings of the 28th International Conference on Machine Learning, ICM, pages lvii–lxiv, 2011.

Y. Wang, D. Tran, and Z. Liao (2011). "Learning hierarchical poselets for human parsing." Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 1705– 1712.

J. Zou, H. Kim (2007), "Using hue, saturation, and value color space for hydraulic excavator idle time analysis", J. Comput. Civ. Eng. 21 238–246.

APPENDIX A: COMPARISON TO PREVIOUS TEST

There is a trade-off between human-intervention, accuracy and computing complexity. The more user annotation the higher the accuracy, and the lower the computation time will be. Figure A.1 and A.2 shows activity recognition accuracy using our method and HMM. Combined with Table A.1, we can conclude that the accuracy of our method is comparable to the HMM, but it uses much lower computation resources and human supervision.

Figure A.1: Confusion Matrix of excavator activity recognition using our method

Figure A.2: Confusion Matrix of excavator activity recognition using HMM

Our experiments show that about 1,000 frames between users annotations are optimal in this trade-off (Table A.1). Table A.1 also compares computation complexity of our method with currently state-of-the-art method of HMM.

Table A.1: Computing efficiency compared with previous State-of-the-art method (per frame)

	Label time	**Detection**	**Tracking**	**Recognition**	**Total**
Our Mtd.	0.1~1 min/label	0.16''	0.03''	10^{-4}''	0.19''
HMM		0.24''	0.06''	0.16''	20.46''

APPENDIX B: GROUND TRUTH FOR ACTION RECOGNITION

Following Table shows the example of the ground truth we manual produced for action recognition evaluation. The activity label is : 1 – swinging, 2 – digging, 3 – dumping, 4 – idling, 5 – moving.

Table B.1: Ground truth example of activity analysis

From Frame No.	To Frame No.	Activity Label
1	571	1
572	732	4
733	832	2
833	1156	3
1157	1243	2
1244	1358	4
1359	1530	2
1531	1812	3
1813	1890	2
1891	2000	4
2001	2101	2
2102	2360	3
2361	2417	2
2418	2538	4
2539	2674	2
2675	2952	3
2953	3018	2
3019	3145	4
3146	3363	2
3364	3631	3
3632	3722	2
3723	3846	4
3847	4039	2
4040	4266	3
4267	4328	2
4329	4399	5
4400	4647	2
4648	8990	1