Hand Tracking by Binary Quadratic Programming and Its Application to Retail Activity Recognition

Hoang Trinh Quanfu Fan Prasad Gabbur Sharath Pankanti
Exploratory Computer Vision Group, IBM T. J. Watson Research Center
trinhh,qfan,pgabbur,sharat@us.ibm.com

Abstract

Substantial ambiguities arise in hand tracking due to issues such as small hand size, deformable hand shapes and similar hand appearances. These issues have greatly limited the capability of current multi-target tracking techniques in hand tracking. As an example, state-of-the-art approaches for people tracking handle indentity switching by exploiting the appearance cues using advanced object detectors. For hand tracking, such approaches will fail due to similar, or even identical hand appearances. The main contribution of our work is a global optimization framework based on binary quadratic programming (BQP) that seamlessly integrates appearance, motion and complex interactions between hands. Our approach effectively handles key challenges such as occlusion, detection failure, identity switching, and robustly tracks both hands in two challenging real-life scenarios: retail surveillance and sign languages. In addition, we demonstrate that an automatic method based on hand trajectory analysis outperforms state-of-the-art on checkout-related activity recognition in grocery stores.

1. Introduction

In this paper, we consider the problem of tracking both hands in challenging real-world environments. This is a challenging problem, with various confounding issues, ranging from low-resolution imaging, occlusions, to rapid hand motion and changing cluttered background. In addition, the problem itself comes with substantial ambiguities in tracking caused by the small hand size, deformable hand shapes and similar or identical hand appearances. In particular, hand ambiguity has greatly limited the capability of current multi-target tracking (MTT) techniques in hand tracking. Current multi-object tracking approaches based on advanced object detectors and appearance models perform very well on benchmark datasets for tracking people, cars and other objects [18, 4, 17, 12]. However, when the appearances of different objects are less discriminant, or even identical, as in tracking both hands, these approaches are prone to issues such as identity switching, and lose tracks. As an example, a state-of-the-art approach for people tracking was introduced in [24] as a recent extension of [4] to account for individual identities. In this work, identity switching is handled by assigning available appearance cues to preserve identities of each track. However, as pointed out by the authors of [24] themselves, identity switching is still a common failure case of their approach, when objects with similar appearances meet and separate.

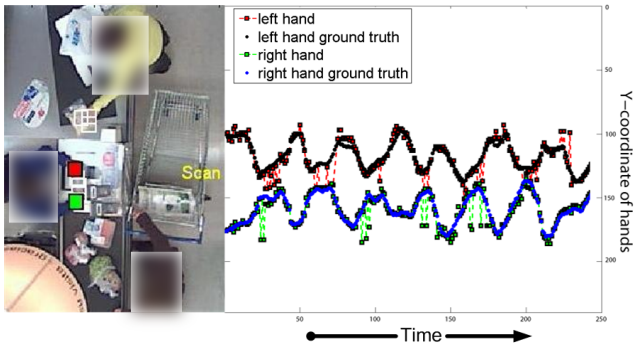

Figure 1. (a) A frame from the retail dataset. (b) Trajectories of both hands projected on the vertical axis are plotted over time along side with ground truth. The hand trajectories are shown to align well with the ground truth.

This problem is even more pronounced in retail. Figure 2 illustrates a frequently occurring case, when the two hands with identical appearances meet and split in reverse directions. Such cases happen extremely frequently, since the two hands meet almost every second, each time the cashier registers an item. Avoiding identity switching in these cases is the key challenge for tracking both hands in the retail scenario.

We believe the key idea to bypass this challenge is to exploit the interaction between hands. Since the two human hands work together in a synchronized way to perform a lot of tasks, they naturally have strong inherent interdependencies. This is contradicting to tracking people, cars,

insects, etc, in which the objects being tracked behave independently, and the inter-object interactions are only limited to mutual occlusions and/or collisions.

Our main contribution is a seamless integration of hand detection, motion model and hand interactions into a novel global optimization framework for hand tracking, using the binary quadratic programming (BQP) formulation. A big advantage of the BQP model is the ability to incorporate any computable pairwise function to model the interactions between hands. Our tracking algorithm automatically finds optimal hand tracks in batches of frames, with no track initialization needed. We validate the effectiveness of our proposed approach on two real-life challenging datasets, i.e. retail surveillance and sign language. Our approach effectively handles key challenges of hand tracking such as occlusion, detection failure, identity switching, and robustly tracks both hands in real-life scenarios. Comparative evaluation shows the superiority of our BQP approach against state-of-the-art approaches.

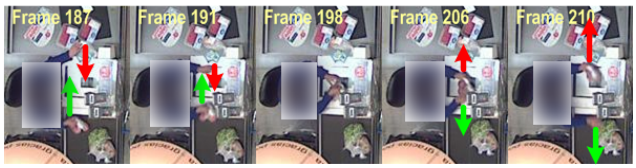

Figure 2. A frequently occurring case in retail: the two hands with identical appearances meet and split in reverse directions. Avoiding identity switching in these cases is a key challenge for tracking both hands.

Hand tracking has found a wide range of applications such as gesture recognition [2], sign language recognition [6] and HCI-based applications [23]. The second contribution of this paper is to explore a novel application in *retail activity recognition*.

The predominant cashier activities at the checkout counter include: *pick-up*, *scan* and *drop*, corresponding to the process of registering one item by the cashier in a transaction. (Figure 3) Many commercial systems including sophisticated machine learning and vision-based approaches have been proposed to recognize such cashier activities for fraud detection [26, 1, 8, 9, 27, 28, 11]. The retail fraud detection problem have gained great attention recently, due to billions of dollars of annual revenue loss in retail worldwide.

Hand trajectories, if available, provide strong cues for cashier activity recognition. For example, the presence or absence of the hand in a specific image region is crucial information to help detect cashier's activities in Figure 3. Moreover, as illustrated in Figure 1, the hand trajectories detected by our proposed approach exhibit strong motion patterns that are highly associated with the checkout activities repeated by the cashier during transactions. As a result, hand tracking leads to a powerful and intuitive method for

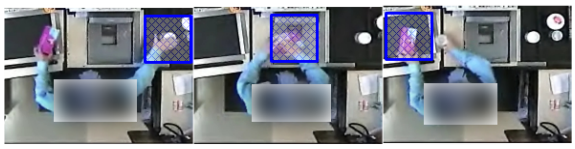

Figure 3. Typical checkout-related activities include: (a) *pick-up*, (b) *scan* and (c) *drop*, corresponding to the process of registering one item by the cashier in a transaction.

recognizing cashier activities under challenging conditions.

We leverage our hand tracking approach and develop an automatic approach to detecting cashier's activities at retail stores (item-pickup, scanning and drop-off) based on analyzing the hand trajectories found by our tracker. Our approach achieves superior performance against the our previous work [9, 27]. Fraudulent events (cashier skipping items on purpose) can then be detected by standard methods, i.e. matching these detected checkout events with the barcode signals, as has been used in other work.

2. Related Work

Hand tracking has long been investigated. Approaches such as [31, 32, 25] focused on tracking single hands based on appearance, sometimes combining with motion models. [19] claimed an approach for tracking both hands, however, this approach still tracked each hand independently, with special attention to cases when the two tracks overlap. Moreover, all these approaches sequentially track from frame to frame, which may easily lead to irrecoverable errors.

A more advanced tracking method is through global optimization over batches of frames, using linear program[20] or dynamic programming[30]. Promising results have recently been demonstrated in a number of such work [14, 4, 17, 3].

One may argue that the BQP formulation that we propose in this paper can be replaced by the simpler linear program (LP). However, as we pointed out earlier, the main contribution of the BQP formulation is the capability to model complex hand interactions, which is the key to tracking both hands effectively. The LP formulation, where only weak linear inter-object constraints (e.g. by L1-norm) are considered, has limited ability to model interactions between hands and therefore will heavily suffer from the pervasive hand ambiguity issue and easily lose track of the hands, or confuse the two hands. As described later in Section 4.1, the superiority of our BQP model is demonstrated by an explicit comparison to the LP approach in [4].

3. Approach

We model the problem of tracking both hands through a sequence of length T as a discrete global optimization prob-

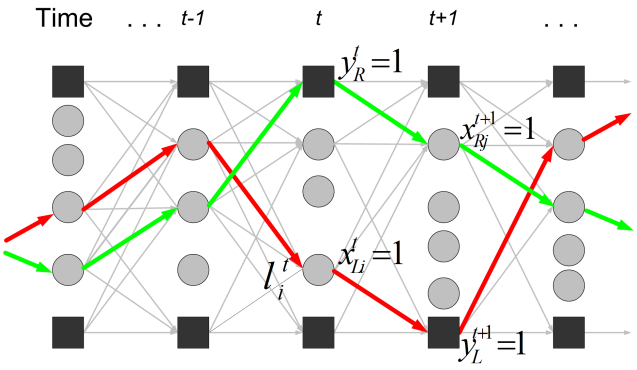

Figure 4. Each column in the graph corresponds to a frame. Gray round nodes correspond to hand candidates detected by the hand detector. Black square nodes represent the *missed* state of two hands. The goal is to identify two most likely disjoint hand trajectories through the frame sequence.

lem as follows:

$$E = \sum_{t=1}^{T} \left(E_D^t + \alpha E_M^t + \beta E_I^t \right) \quad (1)$$

where E_D is the detection cost which is the cost of appointing a detected hand candidate to the true hand. The motion cost E_M enforces the hand dynamic model. The interaction cost E_I models the interactions between the two hands. α and β are weight parameters.

3.1. Tracking Formulation

To convert the above discrete optimization problem to a constrained binary quadratic programming (BQP) problem, we introduce the following binary variables:

- Let $l_i^t = (X_i^t, Y_i^t)$ denote the location of the i^{th} hand candidate at frame t. For each candidate l_i^t, we introduce two binary variables for the two hands: $x_{Li}^t, x_{Ri}^t \in \{0,1\}$. $x_{Li}^t := 1$ appoints the hand candidate to the left hand, $x_{Ri}^t := 1$ appoints it to the right hand.

- To explicitly handle hand missing[1], we introduce 2 additional binary variables y_L^t, y_R^t (for left and right hand respectively). $y_L^t := 1$ indicates that the left hand is missed at frame t (either it is occluded or not detected), $y_L^t := 0$ indicates otherwise.

Figure 4 illustrates a graph representation of the problem. Each column in the graph corresponds to a frame. Gray round nodes correspond to hand candidates detected by the hand detector. Black square nodes represent the *missed* state of two hands. The goal is to identify two most likely disjoint hand trajectories through the frame sequence. A gray node l_i^t is added to the left hand trajectory iff $x_{Li}^t = 1$. A black node y_L^t is added to the left hand trajectory iff $y_L^t = 1$. Exactly one node is added to one trajectory from each column. A node cannot belong to both trajectories.

Our objective is to find:

$$(x^*, y^*) = \underset{x,y}{\operatorname{argmin}}\, E$$

This optimization problem can be formulated as the following BQP problem:

$$E_D^t = \begin{array}{l} \sum_{H\in\{L,R\}} \sum_i x_{Hi}^t \mathcal{C}\left(l_i^t\right) \psi\left(l_i^t\right) \\ +y_H^t \mathcal{C}_{occ}(t) \end{array} \quad (2)$$

$$E_M^t = \begin{array}{l} \sum_{H\in\{L,R\}} [\sum_i \sum_j x_{Hi}^t x_{Hj}^{t-1} f\left(l_i^t, l_j^{t-1}\right) \\ +\sum_j \left(y_H^t x_{Hj}^{t-1}\right) f\left(l_{H*}^t, l_j^{t-1}\right) \\ +\sum_i x_{Hi}^t y_H^{t-1} f\left(l_i^t, l_{H*}^{t-1}\right) \\ +\left(y_H^t y_H^{t-1}\right) f\left(l_{H*}^t, l_{H*}^{t-1}\right)] \end{array} \quad (3)$$

$$E_I^t = \begin{array}{l} \sum_i \sum_j x_{Li}^t x_{Rj}^t \varphi\left(l_i^t, l_j^t\right) \\ +\sum_i x_{Li}^t y_R^t \varphi\left(l_i^t, l_{R*}^t\right) \\ +\sum_i x_{Ri}^t y_L^t \varphi\left(l_i^t, l_{L*}^t\right) \\ +y_L^t y_R^t \varphi\left(l_{L*}^t, l_{R*}^t\right) \end{array} \quad (4)$$

subject to the constraints:

$$\begin{array}{l} \forall t : y_L^t + \sum_i x_{Li}^t = 1 \\ \forall t : y_R^t + \sum_i x_{Ri}^t = 1 \\ \forall t, \forall i \in n_t : x_{Li}^t + x_{Ri}^t \leq 1 \end{array} \quad (5)$$

where

$$\left\langle x_{Li}^t, x_{Ri}^t, y_L^t, y_R^t \right\rangle \in \{0,1\}$$

.

In (5), the first and second constraints guarantee that for each frame, a hand (left or right) must either be assigned to a detected hand candidate or be considered missed, but not both; the third constraint guarantees that no detected hand candidate can be assigned to both hands.

In (3) and (4), we use the notation l_*^t to represent the most likely predicted hand location, in case the hand is declared as missed by the tracker. More details are presented in Section 3.3.

Other notations:

- In Equation (2): $\mathcal{C}\left(l_i^t\right)$ is defined in Equation (6). $\psi\left(l_i^t\right)$ denotes the hand spatial prior, defined in Equation (12). $\mathcal{C}_{occ}(t)$ is the cost of assigning a hand as missed at frame t. We set $\mathcal{C}_{occ}(t) = \frac{1}{|l^t|}$. The idea is that the more hand candidates are present, the less likely the hand is missed.

[1] In the scope of this paper, we will refer to the state of the hand being occluded or missed by detection failure as **missed**.

- In Equation (3): $f\left(l_i^t, l_j^{t-1}\right)$ represents the motion model, defined in Equation (9).
- In Equation (4): $\varphi\left(l_i^t, l_j^t\right)$ models the spatial constraints between hands, defined in Equation (10).

The problem of tracking both hands, after being translated into the BQP form, can be solved using well-known discrete optimization techniques,such as the branch and bound algorithm. Although in theory, BQP is an NP-hard problem, in practice however, certain instances of BQP can be efficiently solved. In our implementation, we used the solver developed by The Hybrid Systems Group at ETH Zurich [2]. The solver actually solved the relaxation of the BQP program using built-in Matlab functions, then a binary solution was found by the branch and bound technique.

3.2. Hand Detection

With the challenges in our dataset such as small hand size, low resolution video, motion blur, it is not suitable to apply sophisticated hand appearance models [13, 21]. The approach in [32] using motion residue cannot be applied either due to the constantly changing background. Here we use a relatively robust and efficient approach for hand detection, based on a hand color model and motion cues, similar to [2] and [27].

Learn the hand color model: Given a training set of hand examples, the hand color model is modeled as a Gaussian $\mathcal{N}(\mu_h, \Sigma_h)$ in RGB space. In retail, we could *adaptively update this model*, since new hand examples are continuously collected from an ROI which is frequently visited by the cashier's hands. (Figure 5(b)). This adaptive model can therefore generalize across different illumination conditions and different cashiers.

Hand candidate detection: A hand probability map is computed for each video frame (Figure 5(a)), by computing the Mahalanobis distance of each pixel from the hand color model. We denote as $p(l_i^t|\mu_h, \Sigma_h)$ the probability of presence of hand at location l_i^t, given the hand color model. This probability is computed by the normalized sum of pixel likelihoods in the square image patch centered at l_i^t. We then define the appearance cost function that measures how well a hand candidate matches the hand color model:

$$\mathcal{C}\left(l_i^t\right) \quad = \quad -\ln p(l_i^t|\mu_h, \Sigma_h) \tag{6}$$

Next, from the computed probability map, we extract $N_t \leq N_{max}$ subwindows with max sum of pixel likelihoods, in which the sum of pixel likelihoods is greater than some threshold. The sum of likelihoods in each subwindow are computed efficiently using integral images. We apply non-max suppression to guarantee that none of the subwindows contain the center of others, though they may overlap. Figure 9 demonstrates our experimental results with different values of N_{max}. Our hand detector is designed to achieve high recall (few misses), at the expense of potentially low precision (many false positives). Although the number of hands in each frame is at most 2, our detector usually detect multiple hand candidates, depicted as white square boxes in Figure 5(b).

[2] http://control.ee.ethz.ch/ hybrid/miqp/

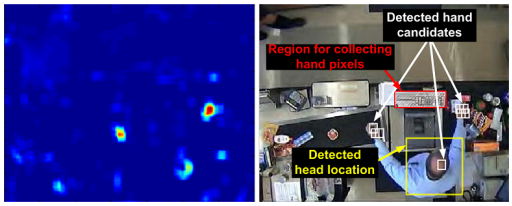

Figure 5. (a) The block-sum hand probability map. (b) White boxes are hand candidate locations detected by our approach, yellow box is the result of our top-down view human detector, the center of which is considered the head location. (*Best viewed in color*)

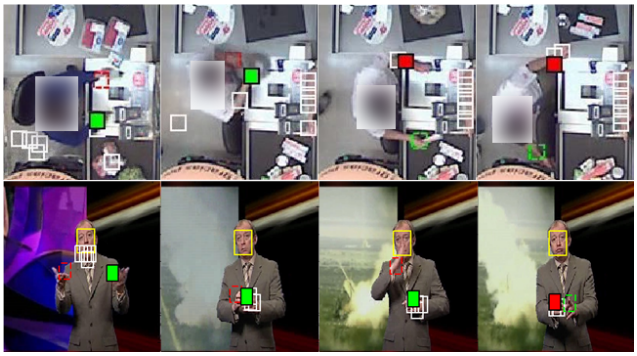

Figure 6. When the hand is missed (due to detection failure or occlusion), our tracker can predict its most likely location at the current frame (unfilled red and green squares).

3.3. Motion Model

We use a linear motion model, in which the *a priori* state estimate $\hat{l}^t$ at time t is modeled as a linear function of the state at the previous frame l^{t-1}, as in Equation (7).

$$\hat{l}^t = Al^{t-1} + w_t \tag{7}$$

$$l^t = \hat{l}^t + K_t\left(z^t - H\hat{l}^t\right) \tag{8}$$

where A is the state transition model, w_t is the process noise with $p(w) \sim \mathcal{N}(0, Q)$. In our implementation, we learn the process covariance Q by maximum likelihood from labeled data. A predicted observation can then be computed as $\hat{z}^t = H\hat{l}^t + v_t$ where v_t is the observation noise, H is the observation model.

Given a new observation z^t (a detected candidate hand location) at time t, we measure the discrepancy between the predicted observation and the actual observation as $r_t = \|z^t - \hat{z}^t\|$, which we call the *residual*. This residual reflects

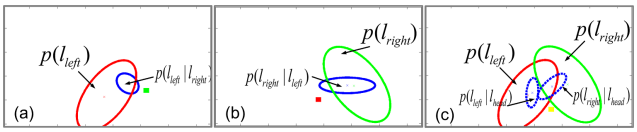

Figure 7. (a) The conditional spatial distribution of the left hand *(blue ellipse)* given the right hand location *(green square)*. (b) The conditional spatial distribution of the right hand *(blue ellipse)* given the left hand location *(red square)*. (c) The conditional spatial distribution of the two hands *(blue ellipses)* given the head location *(yellow square)* compared to the marginal hand spatial distributions *(in green for right hand and red for left hand)*. (*Best viewed in color*).

how well the hand prediction matches the new observation. The probability of the next hand location given the current hand location is defined as:

$$p(l^t|l^{t-1}) \sim e^{-r_t^2}$$

We then set the function $f\left(l_i^t, l_j^{t-1}\right)$ in Equation (3) to be:

$$f\left(l^t, l^{t-1}\right) = r_t^2 \sim -\ln p(l^t|l^{t-1}) \tag{9}$$

The Kalman model is iteratively updated using the standard discrete Kalman Filter algorithm. In cases when the hand is considered missed by the algorithm, our approach can provide predicted locations to these *missed* hands using the *a posteriori* hand estimate l^t computed in 8 (Figure 6).

Note that in cases the hand is missed at frame t and not assigned to any hand location, we assign the hand to the best location found at the previous frame $t-1$, i.e. the hand state with the least residual, denoted as l_*^{t-1}. This notation was used in (3) and (4).

3.4. Modeling Hand Interactions

We model the interactions between two hands using a cost function $\varphi\left(l_i^t, l_j^t\right)$ in Equation (4).

$$\varphi\left(l_i^t, l_j^t\right) \sim -\ln p(l_{left} = l_i^t, l_{right} = l_j^t) \tag{10}$$

where $p(l_{left} = l_i^t, l_{right} = l_j^t)$ is the joint probability of seeing the left hand at l_i^t and the right hand at l_j^t. Using the Bayes rule, we can compute this probability as follows:

$$p(l_{left}, l_{right}) = \frac{1}{2}\left(\begin{array}{l} p(l_{left}|l_{right})p(l_{right}) \\ p(l_{right}|l_{left})p(l_{left}) \end{array} + \right) \tag{11}$$

where $p(l_{left}|l_{right})$, $p(l_{left})$, $p(l_{right}|l_{left})$, $p(l_{right})$ are all modeled using 2D Gaussian distributions for efficiency. We learn these prior and conditional distributions by straightforward maximum likelihood estimation, using labeled training data.

Figure 7(b) and (c) illustrate the strong spatial dependencies between the two hands: the conditional spatial distribution of one hand given the other (i.e.: $p(l_{right}|l_{left})$ and $p(l_{left}|l_{right})$) has much smaller variance than the marginal distribution (i.e: $p(l_{left})$ and $p(l_{right})$).

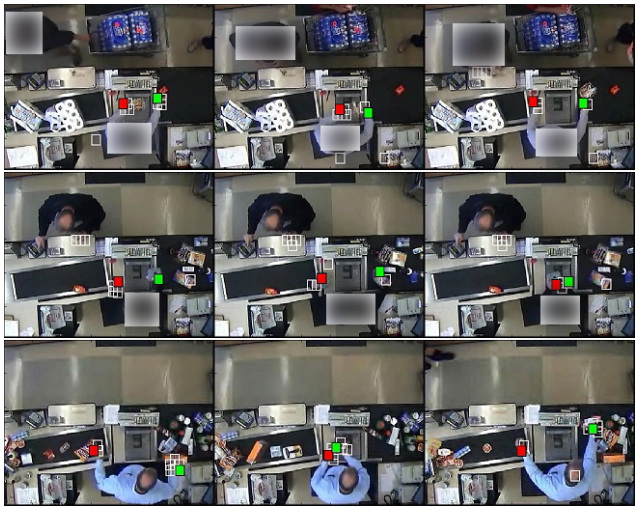

Figure 8. Hand tracking results of our algorithm in the retail dataset. Selected frames from 3 retail sequences.

We also use the head location, when available, to provide additional spatial constraints to hand tracking. Figure 7(c) shows a strong spatial dependency of the two hands on the head location. When head location is detected, we can derive the function $\psi\left(l_i^t\right)$ from the conditional priors $p(l_{left}|l_{head})$ and $p(l_{right}|l_{head})$. Specifically:

$$\psi\left(l_i^t\right) \sim -\ln p(l_{hand}|l_{head}) \tag{12}$$

where $p(l_{hand})$ stands for $p(l_{left})$ or $p(l_{right})$ accordingly. When head location is not detected, the priors $p(l_{left})$ and $p(l_{right})$ are used instead.

One big advantage of writing E_i as an BQP term is the capability to use any computable pairwise function to model the interdependencies between hands. We validate the importance of the hand interactions by experimental results in Section 4.A more complex tree structure such as the pictorial structure [10, 22] can be added to our model if needed, by adding more pairwise constraints. However such an extension will be much more computationally expensive.

Finally, we note that even when one of the two hands is missed, we can still enforce the interactions using the predicted hand (Section 3.3).

4. Experimental Results

4.1. Hand Tracking in Retail Surveillance Videos

Dataset: We evaluate our approach on a dataset of video sequences capturing checkout activities from two real retail stores. This challenging dataset presents large variances in cashiers, backgrounds, camera angles, with significant occlusions, and distractions from customers. The videos are

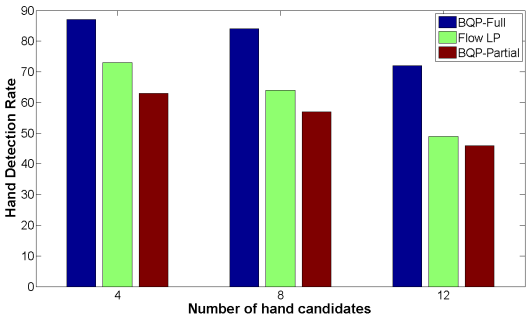

Figure 9. Hand detection rate achieved by our approach (BQP-Full) on the retail dataset and comparison with Flow LP [4], and with BQP-Partial.

at 20 FPS and low resolution (320x240). We manually annotate ground truth square bounding boxes for left and right hands, and heads for these videos. We use a training set of $\sim 1K$ frames, and a test set of $\sim 5K$ frames. There is no overlap between the training set and the test set.

Implementation Details: For our approach (full BQP model), we empirically set $\alpha = 8$ and $\beta = 16$ (in equation (1)). The spatial dependencies between the two hands, and the dependencies between hands and head are learned from training data. We use the hand box size of 15 pixels. We use the approach in [7] to detect the cashier's head from a top-down view camera. Evaluation of our detector with ground truth results in the RMS errors of 7.33 ± 0.23, 10.74 ± 0.29 and 14 ± 0.35 in terms of x, y and shortest distance respectively. An example detection is shown as yellow box in Fig. 5(b).

Evaluation Metrics: We measure the overall hand detection rate (HDR) obtained by tracking both hands. We treat the left hand and the right hand as two different objects for evaluation. The HDR is defined as $HDR = \frac{1}{T}\sum_{t=1}^{T} \delta_\tau(d_L^t, \hat{d}_L^t) + \delta_\tau(d_R^t, \hat{d}_R^t)$ where $\delta_\tau(d, \hat{d}) = \begin{cases} 1 & \left\|d - \hat{d}\right\| \leq \tau \\ 0 & otherwise \end{cases}$ with $\tau = 20$ pixels.

Comparative evaluation: We compare our approach against:

- A state-of-the-art multi-target tracking approach using LP in [4] (Flow LP). We use exactly the implementation from [4], with the neighborhood size of 1, and the possible entrance/exit point(s) being the complete boundary of the detection grid. We note that their method is designed to deal with rather peaky output from detectors, such as the ones produced by state-of-the-art people detectors, as opposed to multiple detections per target, as our hand detector.

- Our own BQP method without the interaction term, i.e. setting $\beta = 0$ in Equation (1) (BQP-Partial). With this change, our model becomes very similar to the LP model in [14], which deals with occlusions, but does not consider inter-object dependencies.

Ground Truth RMS	Hor-RMS	Ver-RMS
14.1 ± 11.3	10.8 ± 8.6	6.9 ± 4.7

Table 1. RMS tracking error in pixels (measuring the average Euclidean distance between the tracking result and the ground truth) over the test set. We also report the RMS in the vertical and horizontal directions.

To this extent, we also tried the Camshift tracking [5] using OpenCV, however Camshift quickly lost track of the hands after a few frames, which yielded its results incomparable.

Figure 9 shows the hand detection rate for each method, with different number of output hand candidates from the detector. We believe the key factor to our superior performance is the ability to effectively handle the identity switch issue, based on modeling the hand interactions. Figure 10 shows several failure cases of the Flow LP approach [4] in our retail dataset. The performance drop of BQP-Partial compared to BQP-Full, by removing the hand interaction term, further substantiates the importance of modeling hand interactions in tracking.

Figure 10. A typical failure case of the LP approach in our retail dataset: the identities of the two hands are switched as they meet and split.

We also measure the RMS errors of our tracking results against the ground truth. Table 1 demonstrates the average RMS errors (in pixels) in terms of Euclidean distance, horizontal distance and vertical distance. The ground truth RMS error of 14.1 pixels is quite acceptable, which is less than the hand box size (15 pixels). Figure 8 displays selected frames from the videos and the hand tracking results. In Figure 11, we plot the hand tracking trajectories of the hands in the horizontal direction across 2 videos, along side with the ground truth.

Running time: On average, with 8 hand candidates per frame, the QP solver takes ~ 7 seconds to compute the hand tracks for 100 frames. (using a Pentium Quadcore 2.53Ghz PC)

4.2. Hand Tracking in Sign Language TV Footage

Dataset: The publicly available dataset [3] of sign language from BBC news consists of $\sim 6K$ continuous frames (first introduced in [6]). Although this dataset includes frequently changing background and self occlusion, it is ar-

[3] http://www.robots.ox.ac.uk/∼vgg/data/sign_language/

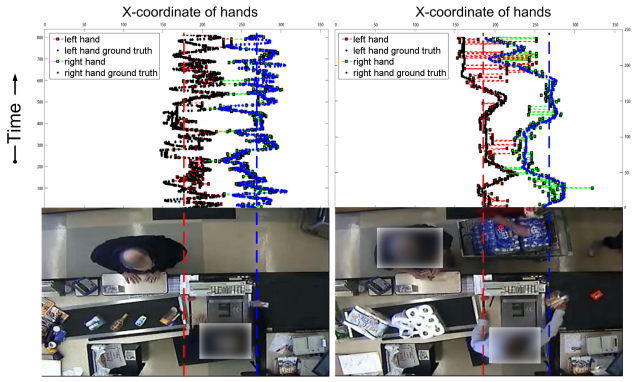

Figure 11. Hand trajectories tracked by our algorithm in two retail sequences. The thresholds for detecting cashier's hand activities are shown by the vertical lines.

Method	**BQP**	[15]	[6]	[16]
Result	**97.1%**	96.8%	95.6%	86.37%

Table 2. Comparison between different methods on the BBC news sign language dataset.

guably less challenging than our retail dataset, due to its higher resolution and high image quality.

Implementation Details: We used the labeled training set to learn the hand color model, the spatial constraints between hands, and between hands and face. We used the Viola-Jones face detector [29] from Open CV to detect faces. For testing, we detected 5 hand candidates per frame, with the hand box size of 30 pixels.

Comparative evaluation: To make our results directly comparable to other state-of-the-art approaches on this dataset, we used the same training set and test set as in the experiment of [6]. We also applied the same metrics as in [6] for quantitative evaluation, i.e., using the overlap measure $o = \frac{GT \cap D}{GT \cup D}$, where GT is the ground truth hand mask, D is our detected hand box, with $o \geq 0.5$.

We evaluated our performance against ground truth, and compared against three state-of-the-art approaches: [6, 16, 15]. Table 2 shows that our tracking approach achieves superior performance to state-of-the-art.

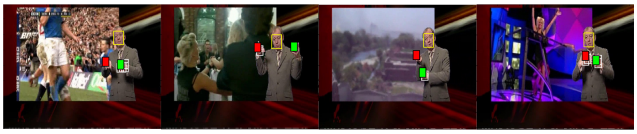

Figure 12. Tracking results on the sign language TV footage.

4.3. Checkout-related Activity Recognition

We leverage our hand tracking results to recognize cashier's activities. We observe that the peaks and valleys of the hand trajectories (as illustrated in Figure 1 and 11) correspond directly to these activities *pick-up*, *scan* and *drop-off*. Specifically, when a pickup event is performed, the cashier's right hand enters then exits the pickup area, which forms a peak in the right hand trajectory. Similarly, a drop event forms a peak in the left hand trajectory. Finally, a scan event forms valleys in both hand trajectories.

Activity	Alg.	Precision	Recall	F-measure
Pickup	BOF[9]	0.84±0.09	0.90±0.04	0.86±0.05
	H-FSM[27]	0.84	0.96	0.90
	Our method	**1.0**	**0.97**	**0.98**
Scan	BOF[9]	0.88±0.06	0.96±0.03	0.92±0.03
	H-FSM[27]	0.83	0.96	0.89
	Our method	**1.0**	**0.84**	**0.91**
Drop	BOF[9]	0.76±0.09	0.90±0.06	0.82±0.07
	H-FSM[27]	0.92	0.86	0.89
	Our method	**1.0**	**0.83**	**0.91**

Table 3. Our performance on retail activity recognition, compared to state-of-the-art approaches using BOF in [9] and using H-FSM in [27].

Based on this observation, we implement an approach for detecting these activities, based on analyzing the hand trajectories found by our tracking algorithm. We first apply the mean filter for smoothing the hand trajectory curves. We then select all the peaks above a certain threshold. These thresholds can be intuitively defined at the borders between ROIs, as illustrated in Figures 11. An example is when the cashier picks up an item, his/her right hand has to cross the border line between the pickup region and scan region twice.

Comparative evaluation: We compare our approach to our two previous methods: an approach using sophisticated learning-based methods in [9], and an approach using hierarchical FSM in [27], using the same videos as in their experiments. For evaluation, we compute the temporal overlap percentage of two activities as in [9], i.e. $\tau = \frac{a_1 \cap a_2}{a_1 \cup a_2}$, with $\tau \geq 0.2$ being the acceptance threshold for each detection. Table 3 reports the detection results from our retail dataset, which shows that our approach achieves better performance on checkout-related activity recognition in most categories, with only one exception being the scan activity recognition slightly worse than [9]. Figure 13 illustrates some examples in which the cashier's activities are correctly recognized by our approach.

5. Conclusion

In this paper, we present a novel approach to tracking both hands, and present a specific application of hand tracking to retail activity recognition. We formulate the two-hand tracking problem as an global optimization problem using binary quadratic programming, where hand detections, motion and interactions are combined together to robustly track both hands in a sequence of frames. Experimental results on two challenging real-world datasets demonstrate the importance of hand interactions in tracking, which is enphasized

Figure 13. Our approach recognizes checkout activities with 100% precision. The recognized activities are annotated in yellow text.

by our BQP approach. We also present an automatic method for cashier activity recognition based on trajectory analysis, which outperforms the state-of-the-art algorithms. In the future, we plan to close the loop between hand tracking and activity/gesture recognition, by incorporating activity-specific information to our tracking model.

References

[1] Agilence. http://www.agilenceinc.com/.

[2] J. Alon, V. Athitsos, Q. Yuan, and S. Sclaroff. A unified framework for gesture recognition and spatiotemporal gesture segmentation. *IEEE PAMI*, 2009.

[3] A. Andriyenko and K. Schindler. Globally optimal multi-target tracking on a hexagonal lattice. In *ECCV*, 2010.

[4] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua. Multiple object tracking using k-shortest paths optimization. *PAMI*, 2011.

[5] G. R. Bradski. Computer vision face tracking for use in a perceptual user interface. In *Intel Technology Journal*, 1998.

[6] P. Buehler, M. Everingham, D. P. Huttenlocher, and A. Zisserman. Long term arm and hand tracking for continuous sign language tv broadcasts. In *BMVC*, 2008.

[7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. 2005.

[8] A. Dynamics. http://www.americandynamics.net/.

[9] Q. Fan, R. Bobbit, Y. Zhai, A. Yanagawa, S. Pankanti, and A. Hampapur. Recognition of repetitive sequential human activity. In *CVPR*, 2009.

[10] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 2005.

[11] P. Gabbur, S. Pankanti, H. Trinh, and Q. Fan. A pattern discovery approach to retail fraud detection. In *ACM SIGKDD*, 2011.

[12] J. F. Henriques, R. Caseiro, and J. Batista. Globally optimal solution to multi-object tracking with merged measurements. In *ICCV*, 2011.

[13] M. Isard and A. Blake. Condensation - conditional density propagation for visual tracking. *IJCV*, 1998.

[14] H. Jiang, S. Fels, and J. J. Little. A linear programming approach for multiple object tracking. In *CVPR*, 2007.

[15] L. Karlinsky, M. Dinerstein, D. Harari, and S. Ullman. The chains model for detecting parts by their context. In *CVPR*, 2010.

[16] M. Kumar, A. Zisserman, and P. Torr. Efficient discriminative learning of parts-based models. In *ICCV*, 2009.

[17] C.-H. Kuo, C. Huang, and R. Nevatia. Multi-target tracking by on-line learned discriminative appearance models. In *CVPR*, 2010.

[18] B. Leibe, K. Schindler, N. Cornelis, and L. V. Gool. Coupled object detection and tracking from static cameras and moving vehicles. *IEEE PAMI*, 2008.

[19] J. Mammen, S. Chaudhuri, and T. Agrawal. Simultaneous tracking of both hands by estimation of erroneous observations. In *BMVC*, 2004.

[20] C. L. Morefield. Application of 0-1 integer programming to multitarget tracking problems. *IEEE Trans. on Automatic Control*, 1977.

[21] E. Ong and R. Bowden. A boosted classifier tree for hand shape detection. In *IEEE Conf. on Face and Gesture*, 2004.

[22] D. Ramanan, D. Forsyth, and A. Zisserman. Tracking people by learning their appearance. *IEEE PAMI*, 2007.

[23] J. Rehg and T. Kanade. Visual tracking of high dof articulated structures: an application to human hand tracking. In *ECCV*, 1994.

[24] H. B. Shitrit, J. Berclaz, F. Fleuret, and P. Fua. Tracking multiple people under global appearance constraints. In *ICCV*, 2011.

[25] B. Stenger, A. Thayananthan, P. H. S. Torr, and R. Cipolla. Model-based hand tracking using a hierarchical bayesian filter. *PAMI*, 2006.

[26] StopLift. http://www.stoplift.com.

[27] H. Trinh, Q. Fan, S. Pankanti, P. Gabbur, J. Pan, and S. Miyazawa. Detecting human activities in retail surveillance using hierarchical finite state machine. In *ICASSP*, 2011.

[28] H. Trinh, S. Pankanti, and Q. Fan. Multimodal ranking for non-compliance detection in retail surveillance. In *WACV*, 2012.

[29] P. Viola and M. Jones. Robust real-time object detection. In *IJCV*, 2001.

[30] J. Wolf, A. Viterbi, and G. Dixson. Finding the best set of k paths through a trellis with application to multitarget tracking. *IEEE Trans. on Aerospace and Electronic Systems*, 1989.

[31] M. Yang and N. Ahuja. Recognizing hand gesture using motion trajectories. In *CVPR*, 1999.

[32] Q. Yuan, S. Sclaroff, and V. Athitsos. Automatic 2d hand tracking in video sequences. In *WACV*, 2005.