Local Depth Patterns for Tracking in Depth Videos

Sari Awwad, Fairouz Hussein, Massimo Piccardi
Faculty of Engineering and IT
University of Technology, Sydney
{Sari.Awwad@student.,Fairouz.Hussein@student.,Massimo.Piccardi@}uts.edu.au

ABSTRACT

Conventional video tracking operates over RGB or grey-level data which contain significant clues for the identification of the targets. While this is often desirable in a video surveillance context, use of video tracking in privacy-sensitive environments such as hospitals and care facilities is often perceived as intrusive. Therefore, in this work we present a tracker that provides effective target tracking based solely on depth data. The proposed tracker is an extension of the popular Struck algorithm which leverages a structural SVM framework for tracking. The main contributions of this work are novel depth features based on local depth patterns and a heuristic for effectively handling occlusions. Experimental results over the challenging Princeton Tracking Benchmark (PTB) dataset report a remarkable accuracy compared to the original Stuck tracker and other state-of-the-art trackers using depth and RGB data.

Categories and Subject Descriptors

I.4.8 [IMAGE PROCESSING AND COMPUTER VISION]: Scene Analysis–depth cues, tracking.

Keywords

Depth videos; privacy; tracking; LDP features; Struck; Princeton Tracking Benchmark dataset.

1. INTRODUCTION AND RELATED WORK

In conventional video surveillance and multimedia applications, video tracking aims to extract detailed trajectory information from individuals' whereabouts. Given that tracking is typically performed on RBG or grey-level data, its natural by-product is rich appearance information about the tracked targets, often permitting their full identification. While this may be desirable in a video surveillance context, it is unacceptable in applications where privacy is paramount such as patient monitoring in hospitals. While it is in principle possible to apply post-processing to obfuscate faces, the availability of appearance data in the first instance poses a latent threat to privacy.

In recent years, the release of sensors such as Microsoft Kinect has made the availability of depth videos inexpensive and widespread.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
MM'15, October 26-30, 2015, Brisbane, Australia.
© 2015 ACM ISBN 978-1-4503-3459-4/15/10 ...$15.00
DOI: http://dx.doi.org/10.1145/2733373.2806295.

A significant trend in tracking research has become the use of depth data in addition to RGB data to disambiguate occlusions and overcome illumination artifacts [2, 13, 16]. However, the possibility to track solely in depth videos has been largely unexplored to date. The challenge posed by depth tracking is major since conventional trackers rely on the targets' appearance and texture to provide correct data association. In this short paper, we investigate the possibility of providing effective tracking based on depth information alone. The motivation for this work comes mainly from hospital environments where it may be important to monitor contacts with patients, hand hygiene episodes, use of instrumentation and others while preventing subjects' identification for privacy reasons.

A tracking algorithm infers the location of a target from a sequence of measurements. In approaching tracking from depth data, the main challenge lies in the design of measurements, i.e., features, effective at tracking single targets through occlusions and in the presence of multiple targets. A common approach to the design of a depth tracker is to make use of skeletal models [15]. However, skeletal tracking is mostly designed to interact with co-operative users and is prone to fail in the presence of major view occlusions. For this reason, in this paper we approach depth tracking by features that do not require fitting of articulated pose models on the target. Our approach consists of extending a proven inferential engine - the Struck tracker of Hare *et al*. [8] - with dedicated local depth features extracted around the target hypothesis. In addition, we introduce heuristic rules leveraging depth information to improve data association. Experiments are carried out over two datasets: a simulated hospital environment dataset collected by these authors, and the recent Princeton Tracking Benchmark (PTB) dataset [16]. The first dataset consists of staged visits to a patient lying on a hospital bed. The second dataset consists of 95 videos varying in target type (humans, animals and non-deformable objects), scene type, presence of occlusion and bounding box distribution. Figure 1 displays samples of depth frames from these two datasets.

Since their inception, consumer depth cameras have found increasing adoption in multimedia and computer vision. For instance, [4] has used depth features in addition to appearance features for improved object recognition; [18] has approached human detection by fitting a 2-D head contour model and a 3-D head surface model over depth data; [9] has used leg history data for detecting human subjects from leg tracks. [6] used depth data to recognise human activities and apply adaptive data compression, winning the 3DLife Grand Challenge at ACM Multimedia 2013. In addition to these works, depth data have found significant use as an additional modality for tracking: [19] leverages point cloud clustering in depth data; [2, 13] use depth-based hierarchical clustering for tracking both individuals and groups; [16] designs dedicated features to resolve occlusions between targets. However, all of these

Figure 1: Depth frame examples from a simulated hospital scenario and the Princeton Tracking Benchmark dataset.

approaches rely on the availability of both appearance and depth data. To the best of our knowledge, this work is the first to attempt general tracking from depth data as the only modality.

2. DEPTH TRACKING

In this section, we provide a brief description of the Struck tracker that is used as the tracking engine, and present the proposed depth feature and occlusion handling procedure.

2.1 The Struck Tracker

The Struck tracker was proposed in [8] as a principled improvement to tracking-by-detection approaches. It leverages the framework of structural SVM [17] to provide a prediction for the displacement of a target from its previous position. By noting the displacement as y and the frame as x, Struck learns a predictive model by the following constrained minimisation:

$$\begin{aligned} &\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i \quad s.t. \\ &w^T\phi(x_i,y_i) - w^T\phi(x_i,y) \geq \Delta(y_i,y) - \xi_i, \\ &\quad i = 1\ldots N,\ \forall y \in Y \end{aligned} \tag{1}$$

The objective in (1) is the standard SVM objective balancing an upper bound over the empirical loss, $\sum_{i=1}^{N}\xi_i$, with a regularisation term, $\|w\|^2/2$. Feature function $\phi(x,y)$ computes a feature vector inside frame x centred on displacement y, and products $w^T\phi(x,y)$ assigns it a score. The constraints in (1) impose that the score assigned to the true displacement of the target, y_i, be greater than that assigned to any other displacement, $y \neq y_i$, by an amount decided by a chosen loss function, $\Delta(y^i,y)$. At its turn, the loss function is set to reflect the overlap between two bounding boxes centred, respectively, on target's location y_i and predicted location y:

$$\Delta(y_i,y) = 1 - overlap(y_i,y) \tag{2}$$

The challenges with the SVM problem in (1) are that the ground truth is unsupervised and that the model requires updating at every new frame. To this aim, Struck uses a number of heuristics to decide on "ground-truth" displacements and which samples to select for the updates [5, 8].

Struck provides three options for feature vector $\phi(x,y)$:

- a 192-D Haar-like feature vector extracted from a grid centred at displacement y;
- a 256-D feature vector of spatially re-scaled raw pixels;
- a 480-D feature vector obtained from the concatenation of 16-bin intensity histograms computed on a four-level pyramid.

2.2 Local depth features for tracking

Many local features have been proposed for tracking in conventional video, including, amongst others, the popular spatio-temporal interest points (SIFT) [11], speeded-up robust features (SURF) [3] and local binary patterns (LBP) [14]. However, local features for tracking in depth video are still a subject of investigation. For this work, we have decided to explore a depth local feature recently proposed for activity recognition in depth video. The feature, called local depth pattern (LDP), resembles LBP in that it computes differences between cells of a local patch [21]. While this feature has proved effective for activity recognition, its performance for tracking cannot be anticipated since these two tasks rely on very different characteristics of the target.

To form our tracking feature (named LDP for tracking, or LDPT for short), we divide the target's bounding box into a $HD \times VD$ grid of LDPs. As values for HD and VD, we typically select 3 and 4, respectively. At its turn, each LDP contains a 3×3 grid of cells. Given that the bounding box has variable size, the size in pixels of the LDP and its cells has to adjust accordingly. The value of each LDP is obtained by concatenating the differences between the average depth of each of its cells with every other. Therefore, the total size of the LDPT feature is:

$$size(LDPT) = HD \times VD \times \binom{3*3}{2} \tag{3}$$

for a total of 432 dimensions. Algorithm 1 shows the detailed steps for computing an LDPT feature.

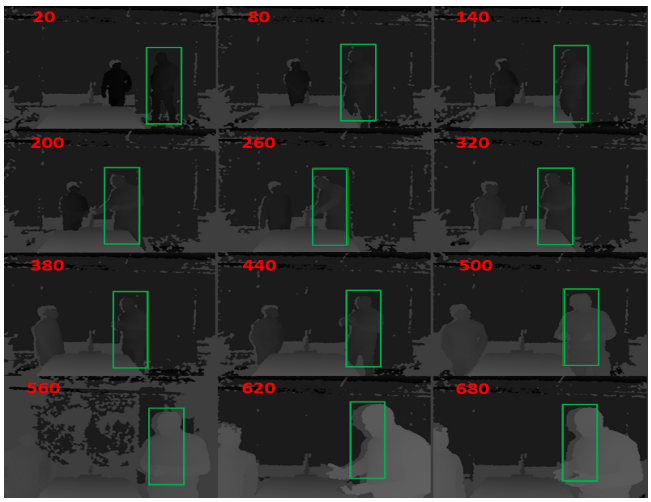
Algorithm 1 The algorithm for computing the LDPT feature.

Input: Bounding box
Output: LDPT feature

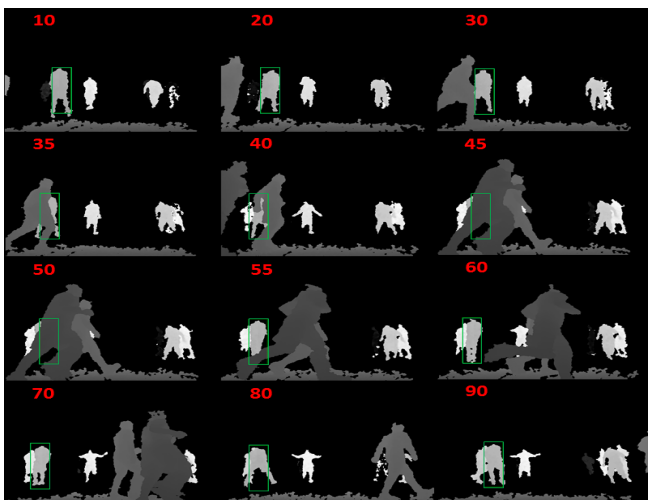
```
{initialises the LDPT feature to an empty set:}
LDPT = ∅
loop r = 1 : VD
    loop c = 1 : HD
        {initialises the LTD(r,c) descriptor to an empty set:}
        LDP(r,c) = ∅
        loop i = 1 : 9
            {loops over all cells in the LTD descriptor}
            loop j = i + 1 : 9
                {computes the difference with every other cell:}
                diff(i,j) = |avgdepth(i) − avgdepth(j)|
                LDP(r,c) = concatenate(LDP(r,c), diff(i,j))
            end loop
        end loop
        LDPT = concatenate(LDP(r,c))
    end loop
end loop
```

2.3 Occlusion Handling

View occlusions from static objects and other targets are likely the main challenge of tracking. While the weakness of depth data is



(a)



(b)

Figure 2: Examples of occlusion handling in a) the hospital simulation and b) PTB datasets.

their lack of appearance features, their strength is the possibility to provide reliable target discrimination based on their distance from the camera. Therefore, in the proposed tracker we have built an occlusion detector that flags an occlusion whenever the depth of the candidate target, d_t, differs from its historical average, d_{avg}, more than a given threshold θ. The measurements are computed at the centre of the respective bounding boxes and the historical average is maintained as a running average of update coefficient λ, updated only in the absence of detected occlusions:

$$occlusion = |d_t - d_{avg}| > \theta \tag{4}$$

$$d_{avg}(updated) = \begin{cases} \lambda d_t + (1-\lambda) d_{avg} & \text{if } occlusion = 0 \\ d_{avg} & \text{otherwise} \end{cases} \tag{5}$$

Figures 2.a and 2.b show examples of successful occlusion handling in a video from our hospital simulation dataset and a challenging basketball video from the PTB dataset. The videos with the full results are publicly downloadable from Dropbox [1].

[1] https://www.dropbox.com/s/8codeji5lnzkg22/hospital.avi?dl=0, https://www.dropbox.com/s/dzuock30489st1u/occlusion.avi?dl=0

3. EXPERIMENTS AND RESULTS

The proposed tracker has been evaluated both qualitatively and quantitatively using a hospital simulation dataset collected by these authors and the recent Princeton Tracking Benchmark (PTB) dataset [16]. Our hospital simulation dataset consists of 26 depth videos staging simulated visits to a patient lying on a hospital bed. These videos are characterised by ample back-and-forth target movement and static occlusions and have been used for qualitative evaluation only [2]. The PTB dataset was released as part of an ICCV 2013 publication to offer a unified, challenging benchmark for tracking in RGB and depth data. It consists of 95 videos varying in target type (humans, animals and substantially rigid objects such as toys and human faces), level of background clutter (plain living rooms, cafes, sport courts etc), and type of occlusions (different durations, appearance changes during occlusions, similarity between targets and occluders etc). The dataset comes accompanied by an evaluation website [3] managed by the benchmark's authors which allows for a blind and unbiased accuracy evaluation. The evaluation protocol considers three types of tracking errors: Type I errors occur when the target is visible, but the tracker's output is far off from the target (wrong detections); Type II errors occur when the target is invisible but the tracker still outputs a bounding box (false detections); Type III errors occur when the target is visible but the tracker fails to produce any output (missed detections). Accuracy figures are divided by target type, target size, movement, occlusion and motion type.

3.1 Experimental Results

Our experiments aim to compare the proposed tracker with the original Struck tracker and other state-of-the art trackers. The qualitative evaluation on the hospital simulation dataset is generally very positive, with the target (a visiting doctor or nurse) successfully tracked in all videos. The original Struck tracker instead tends to lose the target in the presence of large static occlusions.

The quantitative evaluation on the PTB dataset provides us with a test-bed for a rigorous and current performance analysis. Table 1, top part, reports the accuracy comparison for the proposed depth tracker against other trackers using only depth data. These include Struck with different types of features and a tracker based on HOG features [16]. The results in Table 1 show that the proposed tracker outperforms the other trackers in 8 categories out of 11. The inclusion of the occlusion handling module achieves an average accuracy improvement of over 2 percentage points compared to the same tracker without occlusion handling.

The bottom part of Table 1 reports the performance of trackers using RBG data for comparison. The proposed tracker outperforms Struck operating on RGB data in almost every category (10 out of 11). This result is impressive in that it shows that depth tracking with suitable features can outperform RGB tracking at a parity of targets and scenes. In turn, this proves that depth tracking is a viable approach to tracking under privacy-preserving operating conditions. It is also important to add that the performance of Struck on RGB data was reported in [16] as the best out of a pool of popular trackers including TLD [12], CT [20], MIL [1], semi-B [7] and VTD [10]. The only RGB tracker that outperforms our depth tracker in a few categories is the tracker proposed by the authors of the benchmark itself (*OF tracker*, Table 1). Remarked improvements over depth tracking alone is only achieved by fusion of depth and RGB information (*RGBD tracker*, Table 1).

[2] available at https://drive.google.com/file/d/0B9cAe42oTaT_aUs4ckVrazQ1OXM/

[3] http://tracking.cs.princeton.edu/submit.php

Table 1: Accuracy comparison of the proposed tracker with the state of the art on the Princeton Tracking Benchmark.

Algorithm	target type			target size		movement		occlusion		motion type	
	human	**animal**	**rigid**	**large**	**small**	**slow**	**fast**	**yes**	**no**	**passive**	**active**
Struck (Depth videos), Haar	0.31	0.32	0.36	0.29	0.36	0.36	0.32	0.21	0.49	0.36	0.32
Struck (Depth videos), raw pixels	0.34	0.44	0.42	0.37	0.41	0.44	0.37	0.29	0.54	0.43	0.37
Struck (Depth videos), histogram	0.38	0.46	0.44	0.45	0.40	0.51	0.39	0.31	0.57	0.52	0.38
HOG (Depth videos) from [16]	0.43	0.48	**0.56**	0.47	0.50	0.52	0.47	0.38	0.63	0.54	0.48
Proposed tracker (no occlusion handling)	0.39	**0.61**	0.54	0.46	0.51	**0.58**	0.45	0.32	**0.69**	**0.56**	0.46
Proposed tracker	**0.46**	0.59	0.54	**0.52**	**0.52**	0.56	**0.50**	**0.40**	0.68	**0.56**	**0.50**
Struck (RGB videos) from [16]	0.35	0.47	0.53	0.45	0.44	0.58	0.39	0.30	0.64	0.54	0.41
OF tracker (RGB videos) from [16]	0.47	0.47	0.63	0.47	0.47	0.57	0.52	0.47	0.62	0.63	0.49
RGBD tracker (RGB and depth) from [16]	0.74	0.63	0.78	0.78	0.70	0.76	0.72	0.72	0.75	0.70	0.82

4. CONCLUSION

In this paper, we have proposed a novel feature for effective tracking of people in depth videos. The feature, called local depth pattern for tracking (LDPT), extends a recently-proposed feature for activity recognition from depth data. The use of LDPT and the addition of an occlusion handling heuristic to a performing tracking engine, Struck [8], allows achieving remarkable accuracy in depth tracking. The experimental results over the current Princeton Tracking Benchmark show that:

- the lack of appearance information in depth videos is not a final impediment to tracking accuracy. Rather, tracking on depth data can outperform tracking from RGB data at a parity of targets and scene (Table 1);
- the proposed tracker based on the LDPT feature achieves a higher accuracy than existing results on depth data in 8 categories of the benchmark out of 11 (Table 1).

The depth frames displayed in Figs. 1 and 2 give visual evidence that depth data do not disclose identification clues of the targets. This allows adoption of depth tracking in privacy-sensitive contexts such as hospital environments and care facilities.

5. REFERENCES

[1] B. Babenko, Ming-Hsuan Yang, and S. Belongie. Visual tracking with online Multiple Instance Learning. In *CVPR Workshops*, pages 983–990, 2009.

[2] F. Basso, M. Munaro, S. Michieletto, E. Pagello, and E. Menegatti. Fast and robust multi-people tracking from rgb-d data for a mobile robot. In *Intelligent Autonomous Systems 12*, pages 265–276. Springer, 2013.

[3] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110(3):346–359, June 2008.

[4] L. Bo, X. Ren, and D. Fox. Depth kernel descriptors for object recognition. In *IROS*, pages 821–826. IEEE, 2011.

[5] A. Bordes, L. Bottou, P. Gallinari, and J. Weston. Solving multiclass support vector machines with larank. In *24th International Conference on Machine Learning*, ICML '07, pages 89–96, 2007.

[6] S. Chen, P. Xia, and K. Nahrstedt. Activity-aware adaptive compression: A morphing-based frame synthesis application in 3dti. In *21st ACM International Conference on Multimedia*, MM '13, pages 349–352, 2013.

[7] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised on-line boosting for robust tracking. In *10th European Conference on Computer Vision*, ECCV '08, pages I: 234–247, 2008.

[8] S. Hare, A. Saffari, and P. H. Torr. Struck: Structured output tracking with kernels. In *IEEE International Conference on Computer Vision (ICCV)*, pages 263–270, 2011.

[9] S. Koo and D.-S. Kwon. Multiple people tracking from 2d depth data by deterministic spatiotemporal data association. In *RO-MAN*, pages 656–661. IEEE, 2013.

[10] J. Kwon and K. M. Lee. Visual tracking decomposition. In *CVPR*, pages 1269–1276, 2010.

[11] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, Nov. 2004.

[12] M. Luber, L. Spinello, and K. O. Arras. People tracking in rgb-d data with on-line boosted target models. In *IROS*, pages 3844–3849, 2011.

[13] M. Munaro, F. Basso, and E. Menegatti. Tracking people within groups with rgb-d data. In *IROS*, pages 2101–2107. IEEE, 2012.

[14] M. Pietikäinen, G. Zhao, A. Hadid, and T. Ahonen. *Computer Vision Using Local Binary Patterns*. Number 40. Springer, 2011.

[15] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '11, pages 1297–1304, 2011.

[16] S. Song and J. Xiao. Tracking revisited using rgbd camera: Unified benchmark and baselines. In *IEEE International Conference on Computer Vision (ICCV)*, pages 233–240. IEEE, 2013.

[17] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, pages 1453–1484, 2005.

[18] L. Xia, C.-C. Chen, and J. Aggarwal. Human detection using depth information by kinect. In *CVPR Workshops*, pages 15–22. IEEE, 2011.

[19] H. Yang, L. Shao, F. Zheng, L. Wang, and Z. Song. Recent advances and trends in visual tracking: A review. *Neurocomputing*, 74(18):3823–3831, 2011.

[20] K. Zhang, L. Zhang, and M.-H. Yang. Real-time compressive tracking. In *12th European Conference on Computer Vision*, ECCV'12, pages III: 864–877, 2012.

[21] Y. Zhao, Z. Liu, L. Yang, and H. Cheng. Combining rgb and depth map features for human activity recognition. In *APSIPA ASC*, pages 1–4, 2012.