

Real time tracking and detection of non rigid object

AmbarDudhane, RahulRane, KetanPatil, GautamiAher

Abstract: This paper shows that how to detect object using more efficient techniques in real time detection. We successively evaluate the features used in sliding window detection process to decide about object presence/ absence also contain knowledge about object deformation. We exploit these detection features to estimate the object deformation. Estimated deformation is then immediately applied to not yet evaluated features to align them with the observed image data. For increasing the efficiency we only check the windows which are near to the window in which object is detected in previous slide rather than detecting every window everytime. By using sliding window detection process we filter frame and increase tracking speed.

Keywords: Features, classifier, training data, sequence detection, sliding window, object deformation

1 INTRODUCTION

There are lot of algorithms for detecting objects. But it is difficult to find the object when it is non-rigid(moving). It is also difficult to find out the objects which are altered by pose variation. This makes the algorithm slow hence it cannot be used in real time tracking of an algorithm. Therefore it is necessary to create new techniques which can improve the speed of detection of the object. Hence in this paper we have given some techniques which can show us how to improve the efficiency of the algorithm. The tracking and recognition of rigid objects has received a lot of attention in the Computer Vision community, similar tasks for deformable ones. It shows that the successively evaluated features used in a sliding window detection process to decide about object presence/absence also contain knowledge about object deformation. Basically classifier is used to calculate the features. The classifier contains the cluster of images. The sample data contain the positive and negative images. The images of different view-point may be include. Many approaches partition the positive training samples into clusters with similar poses. Some of them first estimate the pose cluster and then use pose-specific classifier to decide about the object presence. Others estimate pose cluster simultaneously with the detection process. A fine partitioning of the pose space is desirable to achieve good detection performance. However, the finer the partitioning, the fewer training samples fall into each cluster and therefore immense training sets are often needed. We show that features evaluated in the sliding window detection process also contain knowledge about the correct alignment of the evaluated window on the observed object deformation. We exploit these detection features to estimate the object deformation. Estimated deformation is then immediately applied to not yet evaluate features to align them with the observed image data. Our approaches partition the positive training samples into clusters with similar poses. Some of them first estimate the pose cluster and then use pose-specific classifier to decide about the object presence. Others estimate pose cluster simultaneously with the detection process. A fine partitioning of the pose space is desirable to achieve good detection performance.

However, the finer the partitioning, the fewer training samples fall into each cluster and therefore immense training sets are often needed. Our feature alignment remedies the partitioning of the training set. The automatic feature alignment keeps the training set less scattered, and improves the detection rates but lacks interpretation. For tacking object we used sequence decision process which speed up the tacking rate. We develop a system which candetect and track non-rigid object in real-time with the high detection and tracking speed. Our system might be used in traffic management, video surveillance etc.

2 RELATED WORK

Great progress has been made in the detection of objects under varying poses and deformations. To train multiple classifiers, either the training data need to be separated into disjoint clusters, or the features in training samples need to be registered to lie in correspondence or both strategies can be combined.

2.1 Detection of deformable objects

Recently, Ali et al. proposed to use pose indexed features coupled with dominant edge orientation estimation, in different scales and positions in the detection window, for feature alignment. By the feature alignment, they forgo the need to train a collection of detectors for different object poses and learn a single deformable detector.

2.2 Search Space Reduction

The search space reduction for detection speed-up has been approached also in Efficient Sub-window Search (ESS). ESS is reducing the search space by a branch and bound algorithm. It defines multiple sets of rectangles (sets of candidate windows) in the image. After evaluation of all the features in the image the algorithm computes the upper bound (highest possible detection score) that the score function could take on any of the rectangles in each set. The authors propose an efficient scheme for going hierarchically through all the possible rectangles (scales and translations) without the need to exhaustively evaluate the detection score for all possible rectangles. Many rectangle sets are rejected as soon as the upper bound is under some acceptance threshold. The disadvantage is the

need for evaluating all the features in the image first. This is well applicable for the approaches which use a bag of features or some shared low level features, usually for multi-view and part-based object detection.

3 SYSTEM FEATURES

3.1 Classifier

Basically classifier is used to calculate the features. The classifier contains the cluster of images. The sample data contain the positive and negative images. The positive images contain main object images and negative images contain other than object images. The images of different view-point may be includes. Many approaches partition the positive training samples into clusters with similar poses. Some of them first estimate the pose cluster and then use pose-specific classifier to decide about the object presence.

The clustering of training data imposes the need to collect large amounts of data for learning each classifier separately. We try to reduce the amount of training data by sharing some features across multiple views to avoid the clustering of training data. To train multiple classifiers, either the training data need to be separated into disjoint clusters or the features in training samples need to be registered to lie in correspondence or both strategies are combined.

3.2 Features matching

The input is in the form of video, so first it has to be converted in the form of frames. The feature of the object to be detected is already stored in the training data. Matching of the training data with the frames take place. The object presence / absence is find out. the successively evaluated features used in a sliding window detection process to decide about object presence/absence also contain knowledge about object deformation. We exploit these detection features to estimate the object deformation. Estimated deformation is then immediately applied to not yet evaluated features to align them with the observed image data. For the alignment estimation we propose regressors that approximate non-linear regression functions and compute the alignment parameters extremely fast. Difficult than the detection of objects in a single pose. If the detection time is con- strained, exhaustive search over the space of possible poses with a single pose detector is intractable. An ample amount of work has been devoted to the detection of objects deformed by pose variations. Our feature alignment remedies the partitioning of the training set. In contrast to which finds simple local deformations (e.g., inplane rotations) and aligns each feature independently, we rather estimate a global non-rigid alignment of all the features. Importantly, the alignment is estimated solely from the features used for detection by pre-trained regressors. The detection of object (e.g. car) is detected using a pre trained data.

3.3 Sequence detection

We get the frames having features detected. By this we can find out the object is present or not. We can also find out the pose variation using pose variation cluster technique. We detect the frame sequentially. So that we

can find out the object deformation by calculating the difference of the same object from the two frames in sequence. By this we can find out the deformation taken place by the object.

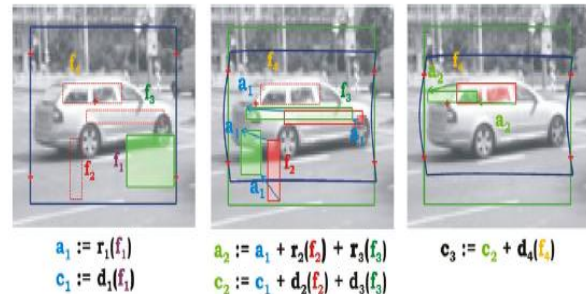


Fig 1

Frame1: In the initial position, only feature 1 is evaluated. From its value the first alignment a_1 and confidence c_1 are computed. Frame2: The alignment a_1 is applied on features 2 and 3. Note that the applied alignment a_2 is updated by contribution of two regressors, not just one. Also note that the alignment a_1 was not applied on feature 4. Frame3: The last feature is moved from its initial position by the accumulated alignment a_2 .

4 ALGORITHM

4.1 Sliding window process

We divide the classification process into K stages. In each stage k , only one feature is evaluated. The value of this feature contributes to the confidence and the alignment. Contributions are determined by (i) a detection Function, $d_k: \mathbb{R} \rightarrow \mathbb{R}$ which maps the feature value to a contribution to the confidence, and (ii) a regression function $r_k: \mathbb{R} \rightarrow \mathbb{R}^m$ which assigns an m -dimensional contribution to the alignment vector a using the same feature value. Both the confidence and the alignment are accumulated from evaluated features. Then there is a threshold $\Theta_k \in \mathbb{R}$ (estimated during the learning), which allows to reject windows with the so far accumulated confidence lower than Θ_k . In each stage, the feature can potentially be aligned. This is determined by a binary value q_k , which is estimated by boosting during the training stage. If q_k is TRUE, this will invoke aligning of the feature, while $q_k = \text{FALSE}$ means that the non-aligned feature will be used.

We keep the last valid alignment, denoted as a_w , where index w corresponds to the stage at which the alignment was estimated. Besides that, we also accumulate alignment updates from all evaluated features. This alignment is updated in each stage k and we denote it by a_k . Hence, there are two alignments: (i) accumulated up to stage k denoted by a_k and (ii) valid, which is applied on features, denoted by a_w . The stage at which the index w is updated is determined by a binary value $z_k = \text{TRUE}$; z_k is again estimated by boosting during the training stage).

Classification of single window by sliding window processes:

1. Initialize $a_0 = 0, c_0 = 0, k = 1, w = 0$.
2. **while** $k < K$ **do**
3. **if** $q_k = \text{TRUE}$ **then** // use alignment

4. Estimate the value of feature $v = f_k(I, a_w)$ with alignment a_w .
5. **else**
6. Estimate the value of feature $v = f_k(I, 0)$ without alignment.
7. **endif**
8. Update confidence $c_k \leftarrow c_{k-1} + d_k(v)$.
9. **if** $c_k < \Theta_k$ **then**
10. reject the window and break,
11. **endif**
12. Estimate alignment $a_k = a_{k-1} + r_k(v)$.
13. **if** $z_k = \text{TRUE}$ **then** // alignment is valid
14. $\text{update } w \leftarrow k$
15. **endif**
16. $k \leftarrow k+1$
17. **endwhile**
18. Accept the window.

7 DISCUSSION

The proposed method shows SDP for sequential detection process. The SDP is favorable for its high detection speed. The regression functions sequentially reduce deformation of the object in the evaluated sub-window. The advantage of the regression functions is that the computational complexity grows only linearly with the dimensionality of the pose space. Accurate regression is usually possible only for a limited range of local deformations.

Main drawbacks of the proposed method are:

- i. Inherently limited generalization of SDP methods,
- ii. Limited range of deformations,
- iii. Key point annotations

Main advantages are:

- i. High detection speed,
- ii. Better detection rate than other SDP methods,
- iii. Global object deformation is estimated as a side product of the detection process.

8 CONCLUSION

We have proposed an efficient approach for aligning detection features with observed non-rigid object deformation in a real time. The idea was shown on sequential decision process, where pre-trained features are successively evaluated in a detection window. The successive feature evaluation allows for efficient alignment estimation by pre-learned regressors during the detection process. The estimated alignment is directly applied to not yet evaluated features which significantly improves the detection rates

FUTURE SCOPE

The proposed project will implement tracking of the object along with detection in java platform. Same can be design on the other operating system if not available. After all, the main aim is to achieve tracking of object in real time. We would like to find the technique to speed up the process. The same project along with artificial intelligence can be used in various applications.

REFERENCES

- [1] K. Ali, F. Fleuret, D. Hasler, and P. Fua, "A Real-Time Deformable Detector," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 34, no. 2, pp. 225-239, Feb. 2012.
- [2] B. Babenko, P. Dollár, Z. Tu, and S. Belongie, "Simultaneous Learning and Alignment: Multi-Instance and Multi-Pose Learning," Proc. European Conf. Computer Vision (ECCV), 2008.
- [3] P. Viola and M. Jones, "Fast Multi-View Face Detection," Technical Report TR2003-96, MERL, 2003.
- [4] J. Zhang, S. Zhou, L. McMillan, and D. Comaniciu, "Joint Real-Time Object Detection and Pose Estimation Using Probabilistic Boosting Network," Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2007