GATE CONNECTED CONVOLUTIONAL NEURAL NETWORK FOR OBJECT TRACKING

KOKUL.T {Kokul.thanikasalam@hdr.qut.edu.au} FOOKES.C SRIDHARAN.S RAMANAN.A PINIDIYAARACHCHI.U.A.J

INTRODUCTION

Visual object tracking is a fundamental task in computer vision. Object tracking has attracted considerable research in the past. However, it is still far from reaching the accuracy of the tracking ability of humans. The objective of this research is to track a single target in a video sequence while only an initial location is provided.

BACKGROUND

Convolutional neural networks (CNNs), have demonstrated state-of-the-art performance in several tasks. However, very few tracking frameworks that have been proposed make use of CNNs because of training data decency.

CONTRIBUTION

Our main contributions are:

1. A gating CNN architecture, which learns generic tracking characteristics by effectively capturing features from front and end convolutional layers.
2. An online domain adaptation mechanism, which is used to train the model with limited samples and reduce over-fitting.

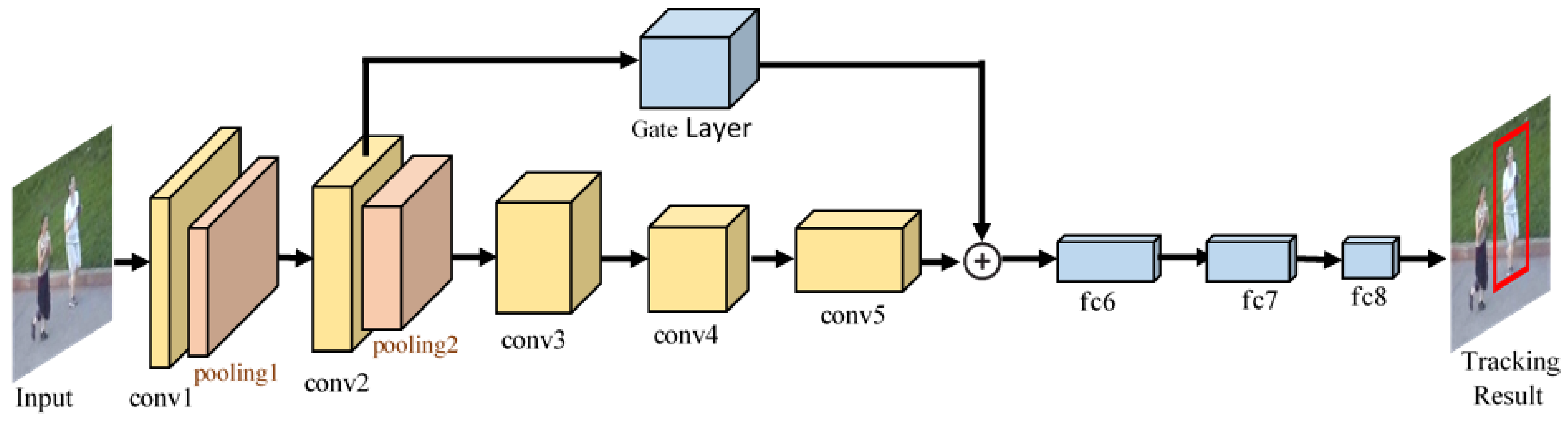
VISUAL TRACKING & CNN

Factors influencing the CNN architecture design for tracking are:

- Deeper networks learn much richer level discriminative features.
- Spatial information is diluted with the depth of the network.
- Early convolutional layer features capture fine grain details and therefore are useful for accurate localisation.
- Last layer features are more effective for inter-class classification.

we propose a new CNN architecture, which effectively learns by fusing the front and bottom layer features of a network to accurately learn target appearance and provide its localisation.

METHODOLOGY: NETWORK ARCHITECTURE



NETWORK DESIGN

- Sequential convolutional layers are obtained from VGG-M.
- Takes input as 96 × 96 and produces output as binary classes (target and background).
- Gate layer,
  - receives input as *conv2* features
  - output is concatenated with *conv5* features through an element-wise addition
  - has kernel size of 3 × 3
- Several gating options are tested in between front and bottom layers and best performed gating is selected.

ONLINE TRAINING & TRACKING

- *fc7* and *fc8* are re-initialised.
- Layers *conv1* to *conv3* are frozen in online training. Remaining are fine-tuned with a lower learning rate, except *fc7* and *fc8*.
- Maximum Mean Discrepancy (MMD) is used to learn the target domain.
- Classification loss and MMD are calculated for each mini batch.
- To locate the target,
  1. A fixed number of samples are collected with different scales and translations around the last known position.
  2. Their corresponding scores are obtained as the network response.
  3. Target location is obtained as the mean of the first five maximum scoring samples.

PRE-TRAINING

- Proposed architecture is trained with a set of annotated video sequences.
- The objective of pre-training is:
  - To train the gate layer to learn the feature mapping between convolutional layers.
  - To train the whole architecture to learn generic tracking characteristics
- During a training iteration, the whole network is trained with samples collected from targets and corresponding backgrounds.

DOMAIN ADAPTATION

Let F^p and F^t be the feature representation at layer l of the pre-trained model and target model, respectively, then the MMD at layer l is,

$$MMD_l(F^p, F^t) = \left\| \sum_{\substack{i=1 \\ f_i^p \in F^p}}^{N^p} \frac{\phi(f_i^p)}{N^p} - \sum_{\substack{j=1 \\ f_j^t \in F^t}}^{N^t} \frac{\phi(f_j^t)}{N^t} \right\|,$$

where,

ϕ - Reproducing features to a kernel space,

N^p, N^t - Number of samples in pre-trained model and target model, respectively.

Target model is learnt by minimizing the loss function,

$$L = L_t + \lambda(MMD^2_{fc7} + MMD^2_{fc8}),$$

where

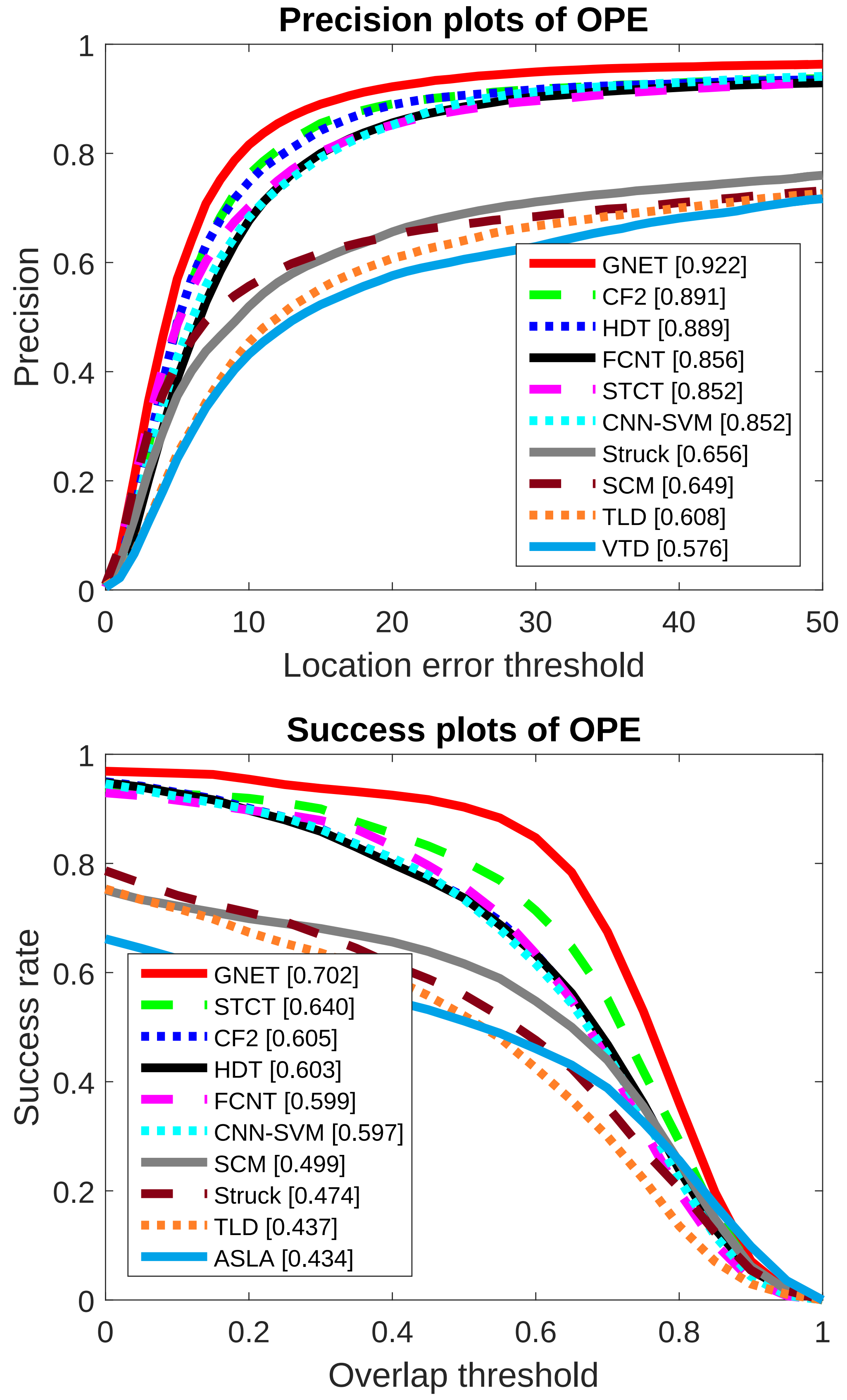
L_t - Logistic classification loss,

λ - Parameter, which is set experimentally.

EXPERIMENTAL SETUP

- Sequences from VOT challenge are used in pre-training (excluding common sequences with test dataset).
- Object tracking benchmark (OTB) dataset is used for testing.

RESULTS



The precision and success plots for the top 10 trackers. The numbers in legend show the representative precision at error threshold of 20 pixels, and the area-under-curve scores for success plots

CONCLUSION

Proposed tracker is evaluated on a publicly available benchmark dataset and demonstrated improved success rate and precision than other state-of-the-art trackers. It runs 1 *fps* on four cores of 2.66 Intel Xeon with NVIDIA Tesla K40 GPU.