Object Tracking with Kernelized Correlation Filters (KCF)

Group Members :
Nan Zhou
Lingzhuo Zhao
Yanjie Chen

Outline

Motivation and Background

- Explore to find a great solution for tracking moving object(s) in different scenarios while maintaining a relatively high speed.

The Problem Definition

- Given a series of video frames, our objective is to track at least one target in the object. It involves researching and developing an algorithm with good performance, testing multiple datasets and performing a precise evaluation.

Method

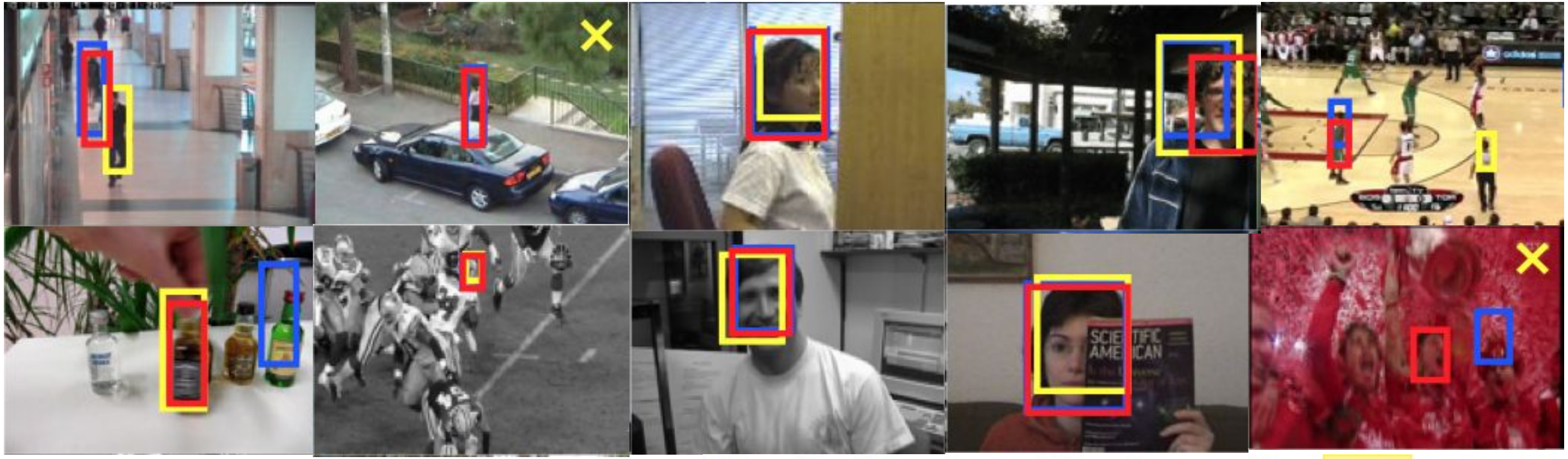
- *Kernelized Correlation Filters (KCF)*

KCF

Kernelized Correlation Filter (KCF):

- Discriminant tracking algorithm
- Based on the idea of traditional correlational filter, it uses kernel trick & circulant matrices to improve computation speed and Histogram of Oriented Gradient (HOG) to boost the performance.

Comparison with TLD & Struck



Kernelized Correlation Filter (proposed) TLD Struck

Key concept:

- ❑ Linear Regression
  - Ridge Regression & closed-form solution
  - Circulant Matrix
- ❑ Non- Linear Regression
  - Kernel Trick
  - Fast Kernel Regression
- ❑ How it works

Ridge Regression

Find a function $f(\mathbf{z}) = \mathbf{w}^T\mathbf{z}$ that minimizes the squared error over samples xi and their regression targets yi.

$$\boxed{\min_{\mathbf{w}} \sum_i (f(\mathbf{x}_i) - y_i)^2 + \lambda \|\mathbf{w}\|^2}$$

$$\boxed{\mathbf{w} = (X^TX + \lambda I)^{-1} X^T\mathbf{y}}$$

λ: regularization parameter

Closed-form solution
where

- *X has one sample per row xi*
- *each element of y is a regression target yi*
- *I is an identity matrix.*

Solution

$$\mathbf{w} = \left(X^T X + \lambda I\right)^{-1} X^T \mathbf{y} \quad ①$$

$$\Downarrow$$

$$\mathbf{w} = \left(X^H X + \lambda I\right)^{-1} X^H \mathbf{y}, \quad ②$$

$$X^H = (X^*)^T.$$

X^H: Hermitian transpose

X^*: complex-conjugate of X

$$X = F \operatorname{diag}(\hat{\mathbf{x}})\, F^H$$

F: constant matrix

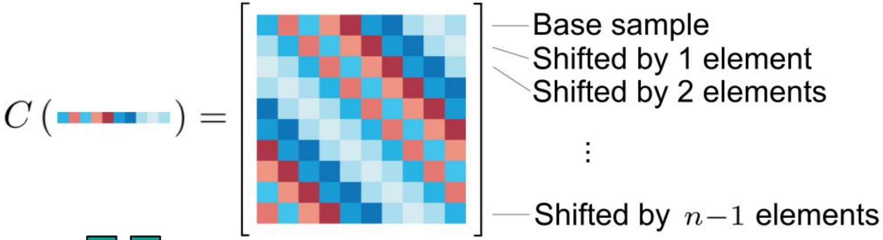
$\hat{\mathbf{x}}$: DFT of the generating vector

$$X^H X = F \operatorname{diag}(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}})\, F^H$$

$$\hat{\mathbf{w}} = \frac{\hat{\mathbf{x}}^* \odot \hat{\mathbf{y}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda}$$

$\odot$: element-wise product

Circulant Matrix



=

$$X = C(\mathbf{x}) = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ x_n & x_1 & x_2 & \cdots & x_{n-1} \\ x_{n-1} & x_n & x_1 & \cdots & x_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_2 & x_3 & x_4 & \cdots & x_1 \end{bmatrix}$$

$$P = \begin{bmatrix} 0 & 0 & 0 & \cdots & 1 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}$$

$$P\mathbf{x} = [x_n, x_1, x_2, \ldots, x_{n-1}]^T$$

$P^u\mathbf{x}$: *u* shifts to achieve a larger translation

Kernel trick

- Expressing the solution *w* as a linear combination of the samples: $$\boxed{w = \sum_i \alpha_i \psi(x_i)}$$

  $\varphi(\mathbf{x})$: non-linear feature space

- Now the goal is to optimize α instead of $\mathbf{w}$.

- Write equation in dot product: $$\boxed{\varphi^T(\mathbf{x})\varphi(\mathbf{x}') = \kappa(\mathbf{x}, \mathbf{x}')}$$

  κ : kernel function (eg. Gaussian or Polynomial)

Fast Kernel Regression

- The solution to the kernelized version of Ridge Regression is given by: $\boldsymbol{\alpha} = (K + \lambda I)^{-1} \mathbf{y}$
- Prove K is circulant for datasets of cyclic shifts. (Satisfied kernels ie. Gaussian, linear, polynomial)
- Diagonalize the equation and obtain: $\hat{\boldsymbol{\alpha}} = \dfrac{\hat{\mathbf{y}}}{\hat{\mathbf{k}}^{\mathbf{xx}} + \lambda}$

  where $\mathbf{k}^{\mathbf{xx}}$ *is the first row of the kernel matrix* $K = C(\mathbf{k}^{\mathbf{xx}})$
- The kernel correlation of two arbitrary vectors, *x* and *x′*, is the vector $\mathbf{k}^{\mathbf{xx}'}$ with elements,

  $$k_i^{\mathbf{xx}'} = \varphi^T(\mathbf{x}')\varphi(P^{i-1}\mathbf{x})$$

How it works

- In the first frame, choose the target object.
- Train a model with the image patch at the initial position of the target. This patch is larger than the target, to provide some context.
- In the following new frames, detect over the patch at the previous position, and the target position is updated to the one that yielded the maximum value.
- Train a new model at the new position, and linearly interpolate the obtained values of α and x with the ones from the previous frame, to provide the tracker with some memory.

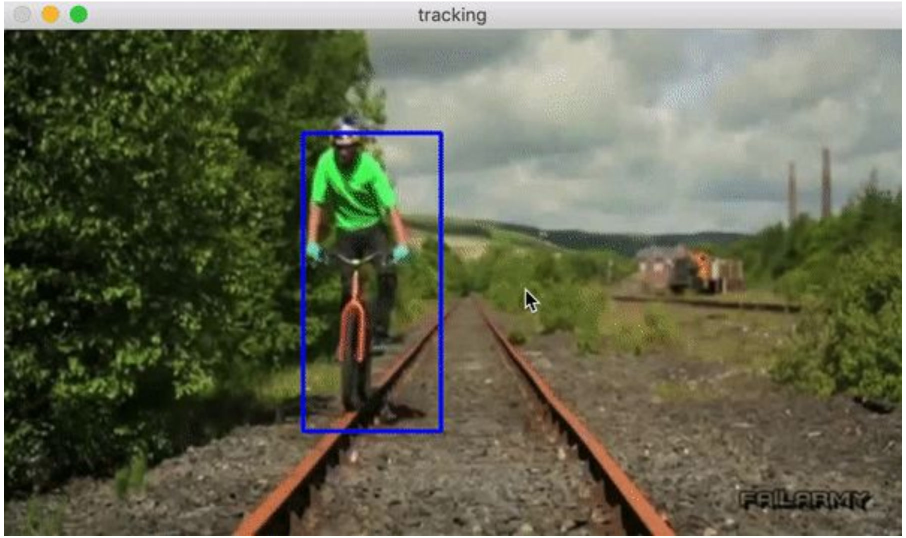
Performance

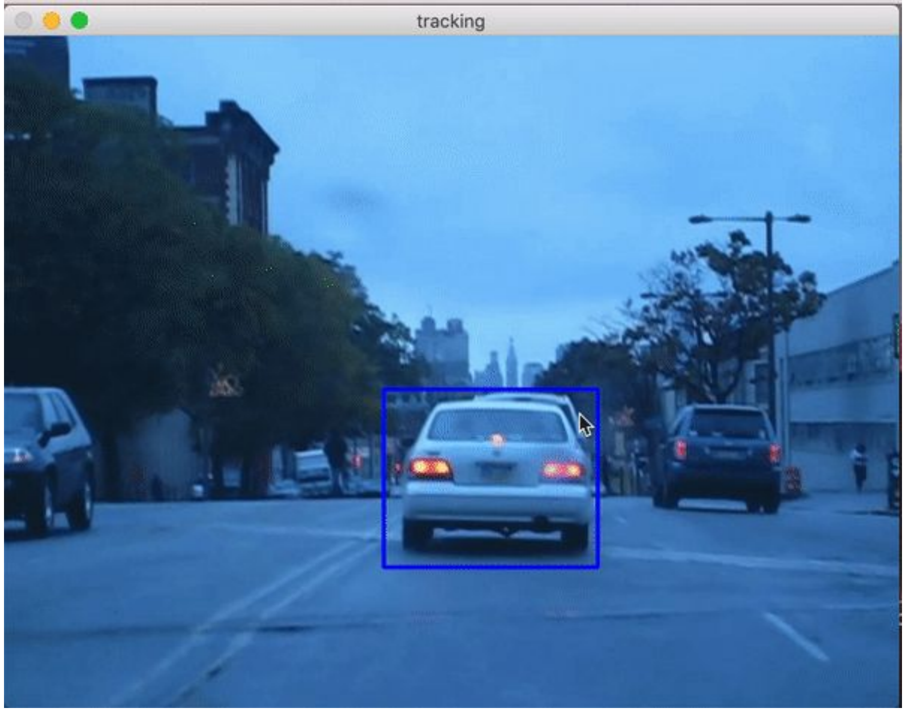
Algorithm	Feature	Mean precision (20 px)
KCF	Raw pixels	*56.0%*
KCF	HOG	***73.2%***
Struck		*65.6%*
TLD		*60.8%*

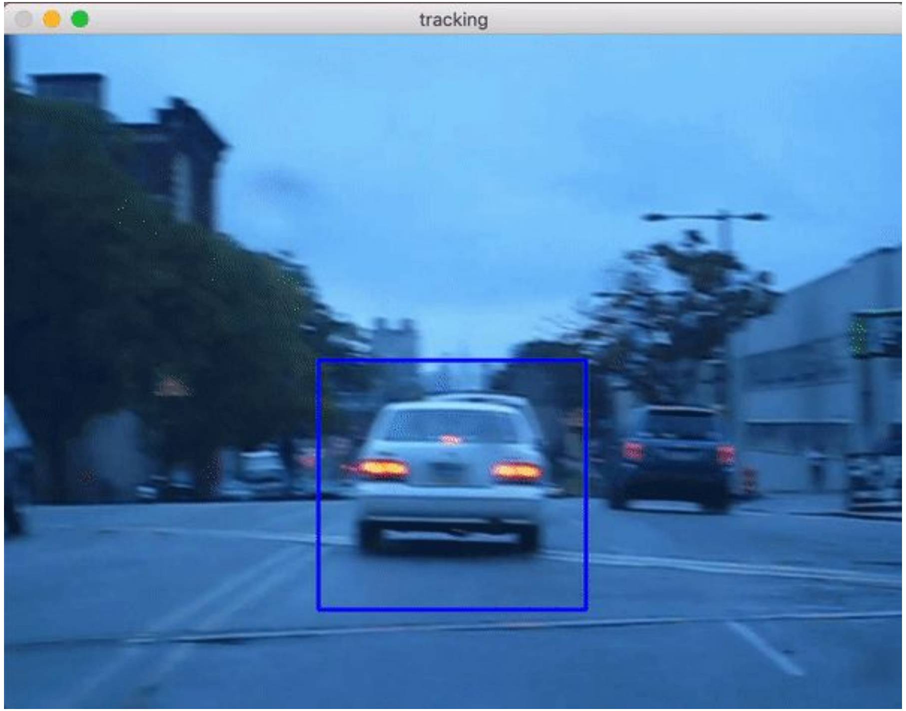
Results

-The initial size of bounding box affects results significantly
-Different dataset behaves differently
-Some parameter need to be adjusted for each dataset


tracking
FAILARMY


tracking
FAILARMY


tracking


tracking

Fail to Track

Discussion and Conclusion

- KCF Being successful in multiple scenarios in both shaking and static camera with objects moving in different directions

- Potential Failure occurs if following exists:
  - Occlusion problem
  - Lost frames
  - Camera zooming causing target object distortion/size change

Future Work

- Certain parameters need to be improved
- Test in multiple datasets
- Develop automatic tracking program without initial bounding box
- Multi-object Tracking

Thank you!

References

1. Henriques, J., Caseiro, R., Martins, P., Batista, J.: Exploiting the circulant structure of tracking-by-detection with kernels. In: ECCV. pp. 702–715 (2012)

2. J. F. Henriques, R. Caseiro, P. Martins and J. Batista, "High-Speed Tracking with Kernelized Correlation Filters," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 3, pp. 583-596, 1 March 2015, doi: 10.1109/TPAMI.2014.2345390.

3. K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang. Fast visual tracking via dense spatio-temporal context learning. In Proceedings of the European Conference on Computer Vision, 2014.

Github Repository And Google Share Link

Github: https://github.com/zlz1996/CS585FinalProject

ShareLink(with GIF):
https://docs.google.com/presentation/d/1LEYoz4B1fPIzQe7gEzXvyj8sr8-WpCa8W4oV2K-60_c/edit?usp=sharing