Seeing the arrow of time

Team AAG

Previous project:

Abstract: A better approach of geometric verification of SIFT features, than RANSAC

Idea:

- Co-linear features are invariant to affine transformation (except rotation)
- Rotate at arbitrary angles and find the WLIS.
- Achieves a bit better performance over ransac

Was changed due to limited scope and limited implementation challenges

Current project title : Seeing the “arrow of time”

Authors of the paper : VGG Oxford, Max Planck Institute, CSAIL MIT, et al.

Objective of paper : Is it possible to tell whether a video is playing forwards or backwards ?

Motivation : To study the temporal statistics of videos, which could be used in

- Video enhancement,
- Filling up of missing data,
- Prediction of what to be seen next, etc.

What is Arrow of time?

- The arrow of time is the "one-way direction" or "asymmetry" of time.
- This concept evolved from second law of thermodynamics.
- This paper proposes different methods to find the arrow of time from video sequences.

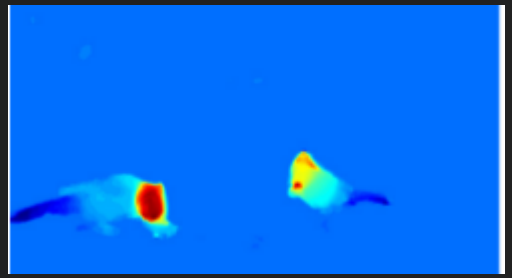
Methods proposed:

1. Statistical Flow
2. Motion Causation
3. Auto Regression Method

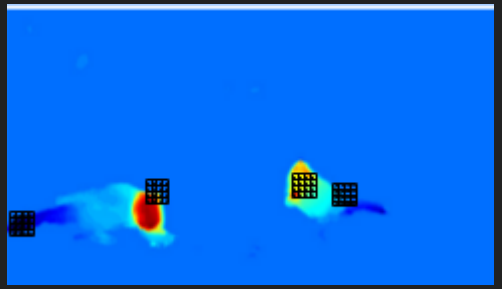
Statistical Flow Method

- **Hypothesis:** Can looking at occurance of particular patterns of motion, give us clues to time of arrow ?

Video frame at “t”



Optical flow using t, t-1

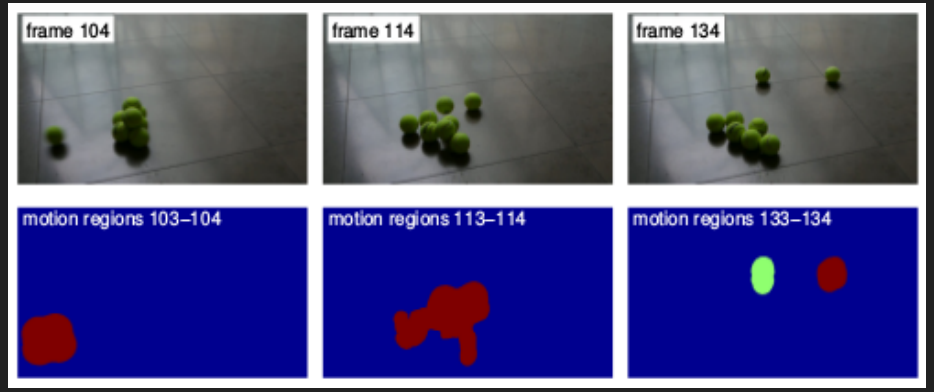


Descriptors

- Descriptors act as temporal statistics, to identify different patterns of motion.

Motion Causation

Hypothesis: One motion can cause multiple motions rather than multiple motions collapsing into one single motion.



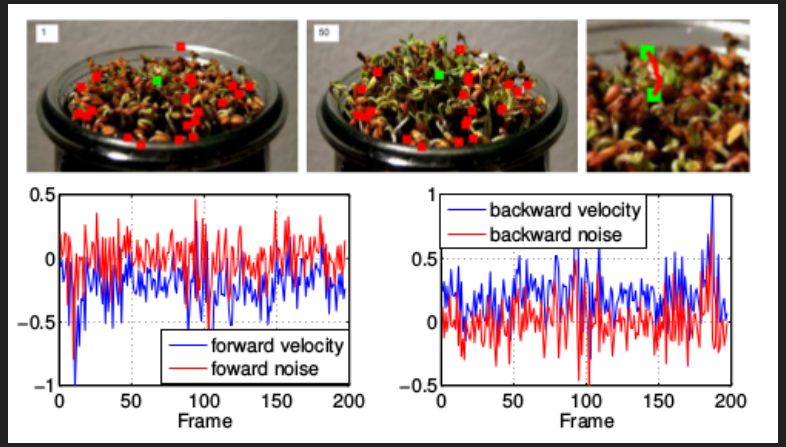
Algorithm:

Calculate moving areas ----> Gauss filters ----> threshold ----> resize ----> find blobs ----> enumerate blobs ---> if blob(t) intersects with blob(t-1) then violation

AR Method

Hypothesis: Deriving the arrow of time using time series analysis, modelling motion velocities through second order AR (auto-regressive) model.

$$X_t = c + \sum_{i=1}^{p} \varphi_i X_{t-i} + \varepsilon_t .$$



Algorithm:

KLT tracker ----> extract velocities of tracked points ----> fit a 2D AR model ---> independence between noise and velocity to determines arrow of time.

Implementation details : Statistical Flow Method

This method is similar to Bag-of-words method in document classification

Implementation Pipeline:

- Create dictionary of descriptors using k-means. => BoW
- Generate descriptors for each video.
- Map each descriptor to the nearest bin. => creates a histogram
- Compute A, B, C, D histograms for each video
- Train supervised classification algorithm and classify on test data

OpenSource Softwares used:

- OpenCV
- Keras
- Jobs run on iiit ada cluster

Results:

Tweaks to original algorithm:

- Modified descriptor generation. Instead of sift-like descriptors we used a softmax over neighbourhood.
- Used PCA to compress the histograms of each video.
- Used 3 layer MLP along with SVM to compare the performance.

Published accuracy: 90% on Train:Test => 120:60
MLP accuracy: 99.29 % on Train:Test => 70:60
SVM accuracy: 100% on Train:Test => 70:60

Dataset:

Custom Youtube dataset created by the publishers.
Train dataset: 70 videos
Test dataset: 60 videos