Video Tracking of Small Unmanned Aerial Vehicle

Steven Krukowski, Adrien Perkins

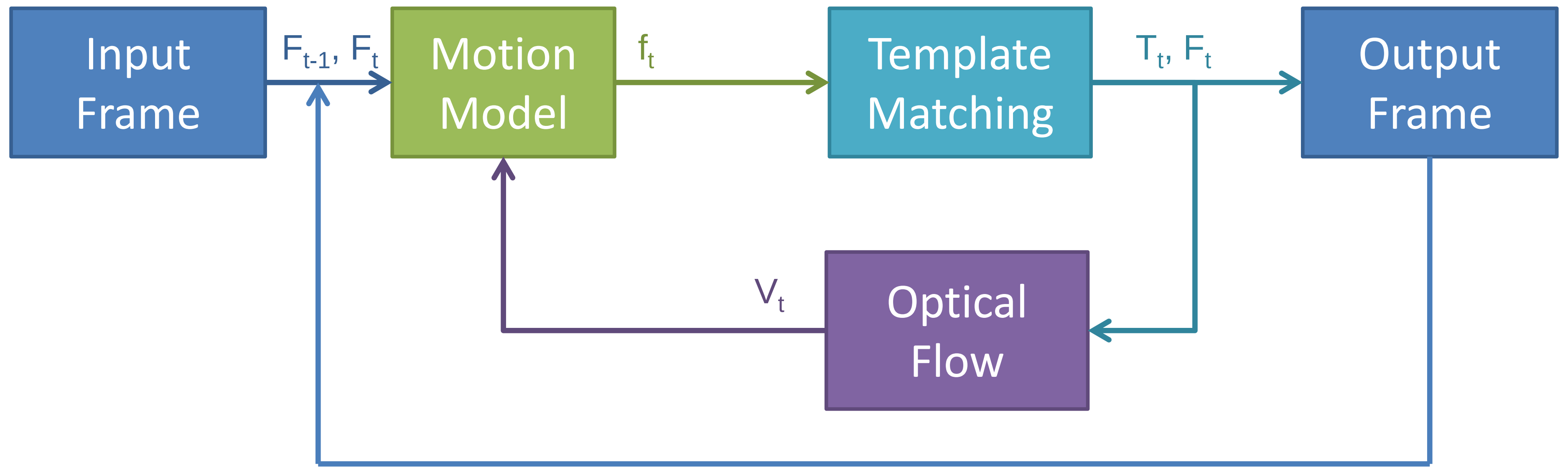
Department of Aeronautics and Astronautics, Stanford University

Motivation

Filming of UAV with a smartphone is a daunting task given the size of UAV, especially at a distance. This projects sets out to assist a user by tracking a small UAV in a live video stream on android using OpenCV.

Can you spot the UAV?

Tracking Technique



Video Frame
F_t = frame time t (current)
F_{t-1} = frame time t-1 (previous)

Optical Flow
V_t = target's velocity time t
V_{t-1} = target's velocity time t-1

Template Matching
f_t = search area time t
f_{t-1} = search area time t-1
T_t = template time t
T_{t-1} = template time t-1

Algorithm Overview

0. User selects initial frame
1. Get new frame
2. Motion model to predict location of target
3. Template match over predicted location
4. Determine velocity of target using optical flow for next time step – back to 1.

Motion Model

$$i' = i + \delta i + \epsilon$$
$$j' = j + \delta j + \epsilon$$

- ✓ Estimated template position change
- ✓ Leverage known information about tracking target
- ✓ Used to determine search window within frame
- ✓ Reduce false positives by ignoring other areas of frame
- ✓ Speed up processing time to allow for real time template matching
- ✓ Can be estimated using template matching target motion in image frame
- ✓ More refined from optical flow results

Optical Flow

- ✓ Lucas Kanade OpenCV implementation
- ✓ Calculate object and background velocity
- ✓ Used to update the motion model

Template Matching

Motion Update & Weighting
- Use motion model to find probability of next template location

Template Match
- Gaussian model template matching on the narrowed search area

Total Weight
- Final result reduces false positive rate with motion weight

Update Template
- Use max likelihood estimate template location in next frame

Motion Update Model:

$$P(i',j'|i,j,\delta i,\delta j) \propto e^{\frac{-\left((i'-i)^2+(j'-j)^2\right)}{\sigma_v^2}}$$

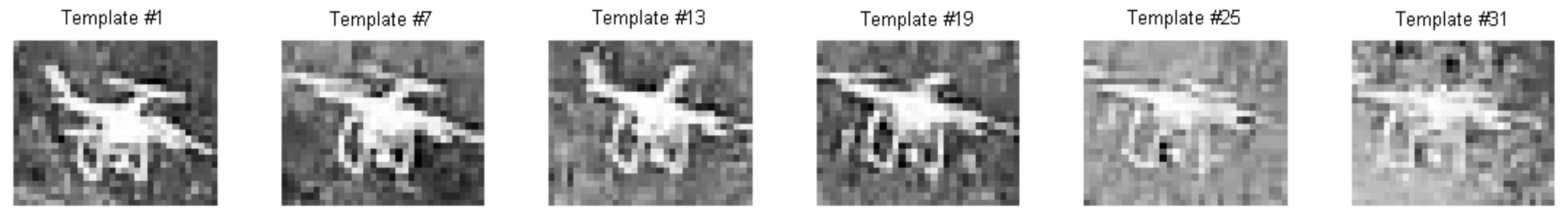
Gaussian Template Matching Model:

$$P(i'',j''|i',j',T_{t-1},F_t) \propto e^{\frac{\Sigma_{i,j}\left(F_t(i+i'',j+j'')-T_{t-1}(i,j)\right)^2}{\sigma}}$$

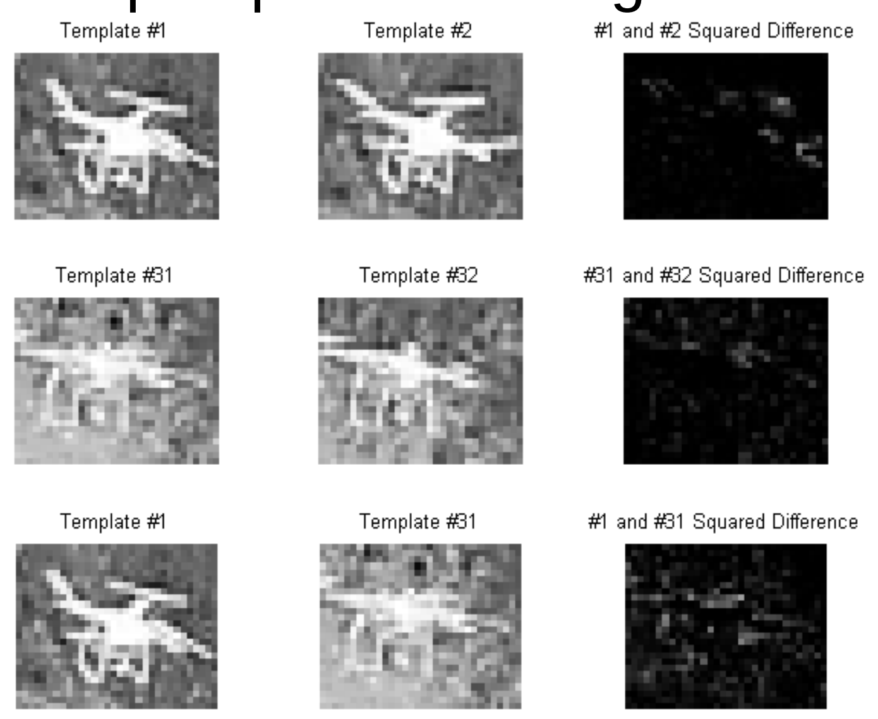
T: *Normalized mean removed template*
I': *Normalized mean removed search frame*

Why update the template?

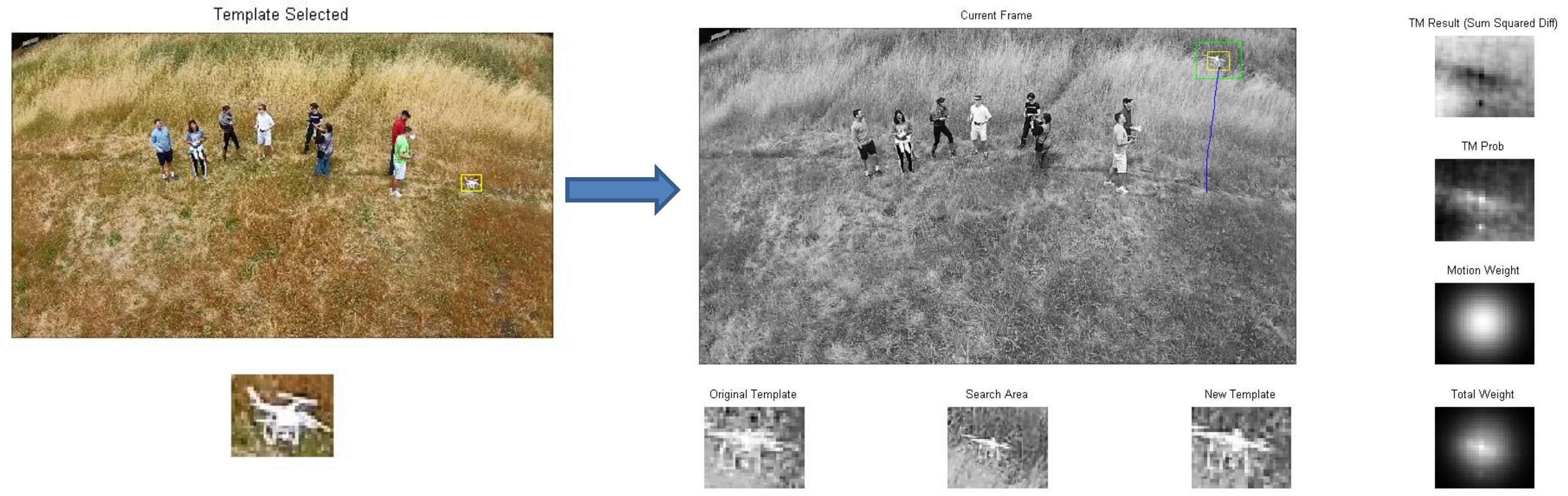
Significant background changes effects template matching



Vehicle perspective changes over time


Template #1
Template #2
#1 and #2 Squared Difference
Template #31
Template #32
#31 and #32 Squared Difference
Template #1
Template #31
#1 and #31 Squared Difference

Experimental Results



Android Live Vide Runtimes

Method	Search Area	Approximate Frame Rate (fps)
Optical Flow	Entire image	3-4
Optical Flow	Motion limited	13-15
Template Matching	Entire image	3-4
Template Matching	Motion limited	10-12
Optical Flow + Template Matching	Motion limited	10+

Future

Live video downlink with tracking