GENERALVISION

Application Note

TARGET TRACKING WITH NEUROMEM

The essence of target tracking is to recognize an initial target, recognize it as high speed and be able to relearn its appearance if any drift occurs due to a change of trajectory, partial obstruction, etc.

NeuroMem is the only neuromorphic chip capable of high-speed recognition and autonomous re-learning capabilities in a footprint of a few square millimeter and for a power consumption of mWatts.

View videos on GV youtube account

- Car tracking with fixed camera
- Face tracking with fixed camera
- Air Plane tracking with motorized camera
- Stereo Tracking with two motorized cameras
- Other

THE APPLICATION

Target Identification

Option A: Manual selection

Option B: Automatic novelty detection

Option C: Identification using a predefined knowledge

Output the inital target location

Tracking with Fixed camera

Adaptive Region Of Search (ROS) around last target position.

Build new local knowledge of the target at different close positions.

Output of the neurons during the search is used to calculate the target position.

Re-inforce knowledge if confidence of the neurons decreases by learning new models.

Tracking with Motorized camera

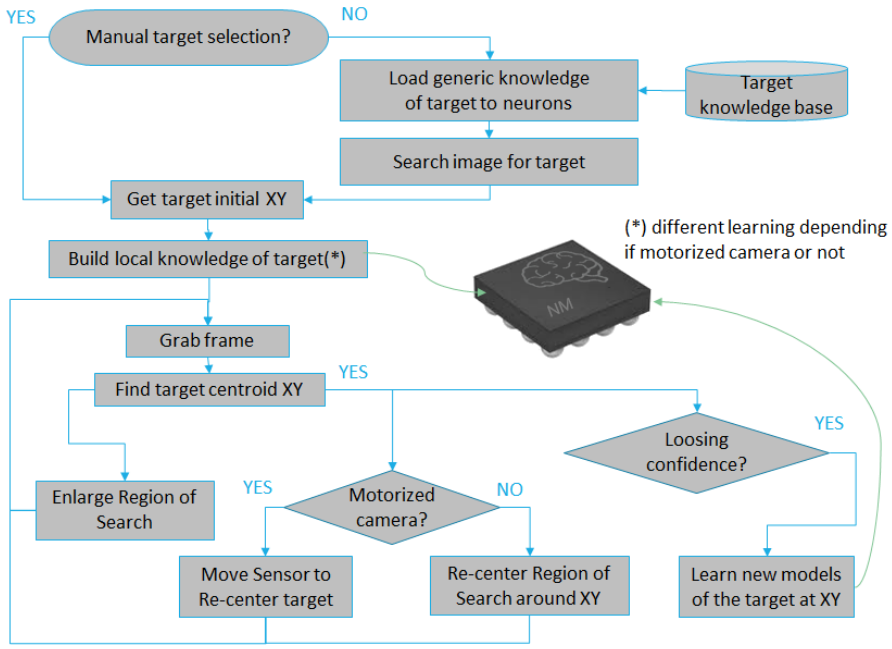
Fixed Region of Search (ROS) around initial target position

Build new knowledge to represent the intial target at different cardinal offsets.

Output of the firing neurons is used to determine a "motion correction" for the camera.

Re-inforce knowledge if confidence of the neurons decreases by learning new models.

FUNCTIONAL OVERVIEW



STEP 1: TARGET IDENTIFICATION AND INITIAL LOCATION

The initial target can be selected manually, or it can be identified using a knowledge built at an earlier time by the neurons.

In the latter case, a search over the image will produce the list of (X,Y) positions where the target is recognized. If you are using the CogniSight SDK, the FindROSObjects function will give you this list and the L1 distance of the closest firing neuron for each identified location. The initial target location can be calculated as the centroid of these (X,Y) positions. Depending on your application, you may want to use the L1 distance to filter out some of these (X,Y) positions out. Indeed, the higher the distance, the less similarity between the incoming pixels and the model stored in the neuron.

STEP 2, TRACKING WITH FIXED CAMERA

- Build a new local knowledge of the target including adjacent and overlapping models
- Continuously scan the Region Of Search (ROS) around the last target position and calculate the new target position in the current frame
- Adjust the next ROS location based on target position and trajectory. Enlarge ROS if target is lost
- Re-inforce knowledge by learning new models if the overall confidence of firing neurons decreases

BUILD A LOCAL KNOWLEDGE OF THE TARGET

1) Set the size of the Region of Interest (ROI) to the minimum area containing discriminant information about the target. For example, to track an airplane, do not set the target size to include the entire airplane and consequently lots of sky area with the variability it can contain such as clouds, sunshine, etc. Instead learn its tail, the tip of a wing, etc. The benefits of a small target size are: (1) Faster feature extraction, (2) Less contextual variability, (3) Less sensitivity to changes in scale and orientation

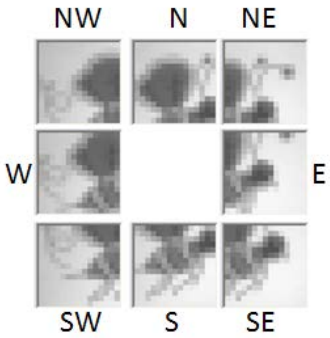
2) Learn the target at its initial location as category “target”. Note that the neuron committed after this operation can have a large influence field and over-generalize easily. This influence field will actually be set to current value of the Maximum Influence Field (MAXIF) when the first example of the target is taught.

3) Sample adjacent neighbors sampled at D pixels from the centered target and for each of them, learn it with a category 0, and immediately re-learn it with a category “target”. The first Learn will have the effect to shrink the influence field of the neuron holding the centered target, while the second learn will add the model of the neighbor to the knowledge under the same category “target”. This mechanism forces the commitment of multiple “conservative” neurons representing the target. Without learning the category 0 first, there is a good chance that the 1st neuron will always recognize the neighbor and prevent the addition

of tis model to the knowledge. The distance D can be a fraction of the target size and measured from the center of the selected target.

Depending on your application and the type of motion of the target, the adjacent neighbors can be the following subsets:

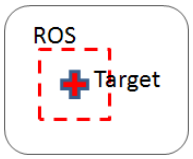
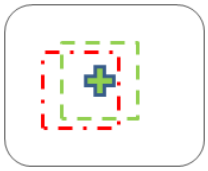
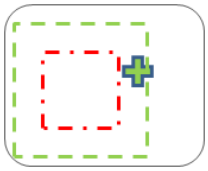
- (W, E), if target is known to move only vertically
- (N, S), if target is known to move only horizontally
- (NW, SE) or (NE, SW)
- (N,E,S,W) or (NW, NE, SE, SW)
- (NW, N, NE, E, SE, S, SW, W)



4) Also depending on the application, you could teach the central target and its adjacent neighbors under different orientation and different lighting (manually or programmatically)

TRACK

Tracking is obtained by moving the ROI over a limited Region of Search (ROS) and monitoring the positions of the ROI which are recognized by the neurons. The tracking engine can calculate their center of gravity (possibly filtering false and isolated hits). From this position and the direction of the last displacement, it can anticipate the Region of Search in the next frame.

Frame T		Frame T+1
Image  	Case #1: If target is detected at a different location but still inside the ROS (red outline) ➔ Move(*) the ROS so it will still contain the target in the next frame (green outline)	
	Case #2: If target is not found inside the ROS (red outline) ➔ Enlarge(**) and/or move(*) the ROS so it will still contain the target in the next frame (green outline) Repeat at every frame until target is found, then go back to Case #1	

(*) The trajectory of the target helps choose the best next move which can be to re-center the ROS at the last target position, or rather off-center it in the direction followed by the target.

(**) Increasing the size of the ROS also increases the search time and therefore reduce the chances to lock on a fast-moving target. It is recommended to increase the ROS the least and rather move it to a new location with higher probability to contain the target based on its trajectory in the past frames.

The search pattern used to look for the target within the ROS can be critical. For example, a raster scan from upper-left to lower-right will be suitable for a target moving in the opposite direction, that is from the lower-right to the upper-left. Indeed, the target should be detected at the beginning of the scanning in the next frame.

STEP2, TRACKING WITH MOTORIZED CAMERA

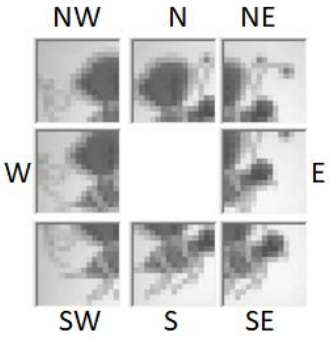
- Fixed Region of Search around initial target position
- New knowledge is set to representation of the initial target at different cardinal offsets
- Output of the firing neurons is used to determine a “motion correction” for the camera

LEARN

1) Set the size of the Region of Interest (ROI) to the minimum area containing discriminant information about the target. For example, to track an airplane, do not set the target size to include the entire airplane and consequently lots of sky area with the variability it can contain such as clouds, sunshine, etc. Instead learn its tail, the tip of a wing, etc. The benefits of a small target size are: (1) Faster feature extraction, (2) Less contextual variability, (3) Less sensitivity to changes in scale and orientation

2) Learn the initial target as category “Center”

3) Learn the adjacent neighbors with the categories NW, N, NE, E, SE, S, SW, W



Neighbors can be selected at a distance D = 1/8, ¼, ½ of the initial target center

Neighbors can be selected at limited positions depending on the target motion

- (W, E) if horizontal motion only
- (N, S) if vertical motion only

TRACK

ROS is fixed and centered around the initial target position

At each frame, raster or spiral search within the ROS and build a histogram of the recognized cardinal categories. You could decide to weigh this histogram with the L1 distance values which represent the inverse of a confidence factor.

The dominant category gives the inverse corrective motion to be applied to the camera, so the target is brought back to the center of the ROS

REINFORCE LEARNING

In addition to monitoring the “firing” locations, the engine can also monitor the confidence level of the firing neurons which is inversely proportional to their firing “distance”. Indeed, if the sum of the distance of the firing neurons starts increasing, one can suspect that the target is starting to drift from the models which are stored in the neurons. It might be time to force the neurons to learn a new model of the target before it can no longer be recognized.

Re-inforce learning is very time sensitive. You want to make sure the decision is made within a time frame where the target is indeed at the last recognized X,Y location. Otherwise, you may add models to your knowledge which actually do not contain the target and under such circumstances the knowledge will become corrupted.