EE692 Project

Unmanned Wildlife Surveillance

Presented by Amit Thomas, Joshith Gubbala and Putta Sravankumar Reddy

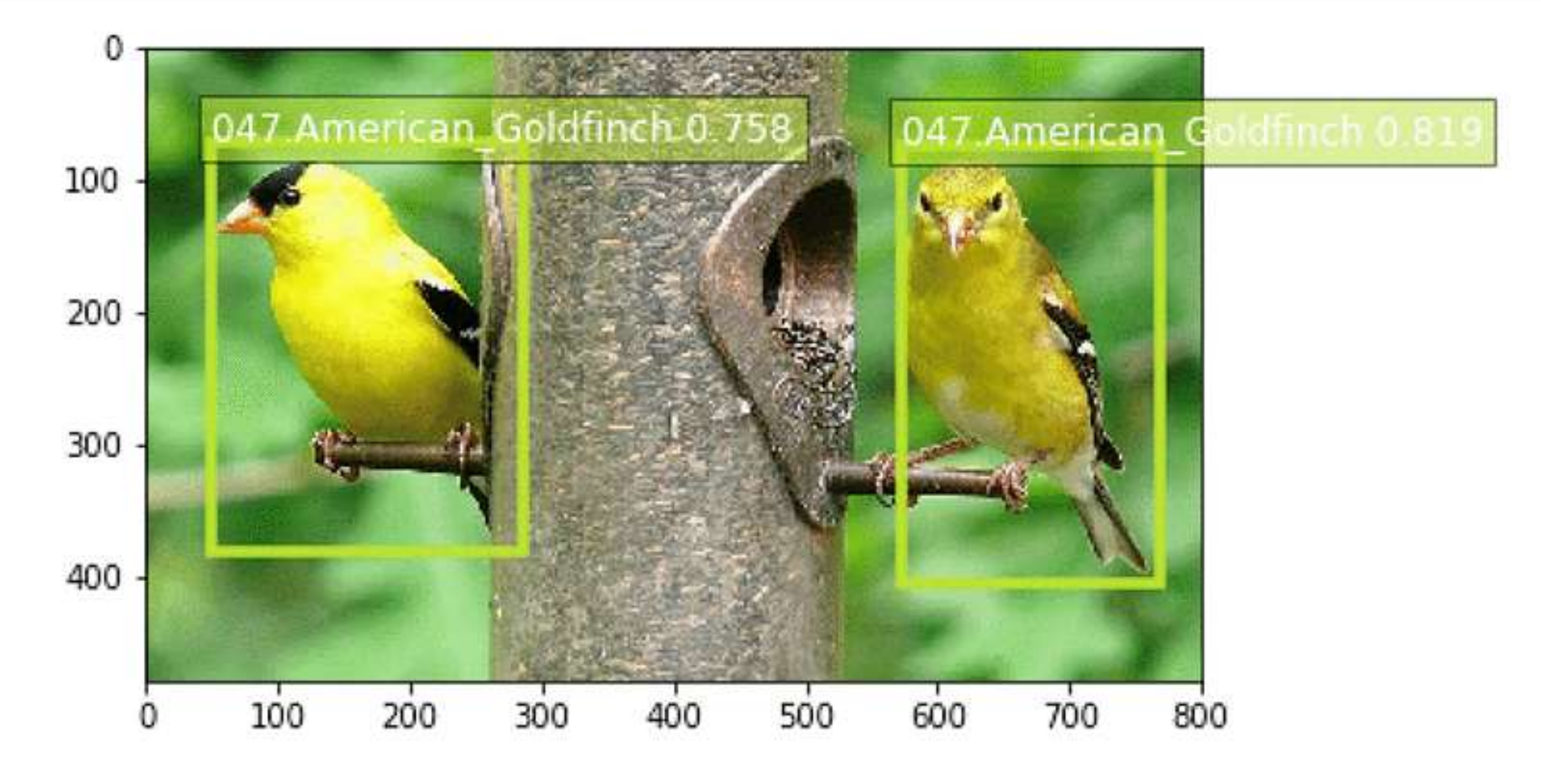
Background

- Wildlife photography involves a lot of waiting and focus
- Static capture suitable for automation
- Current on-site implementations include camera traps and live video surveillance
- Camera traps have motion sensors that detect change in environment which upon triggering capture frames
- Live video surveillance implementations stream the video feed to a server

Problem Explanation

Level 1

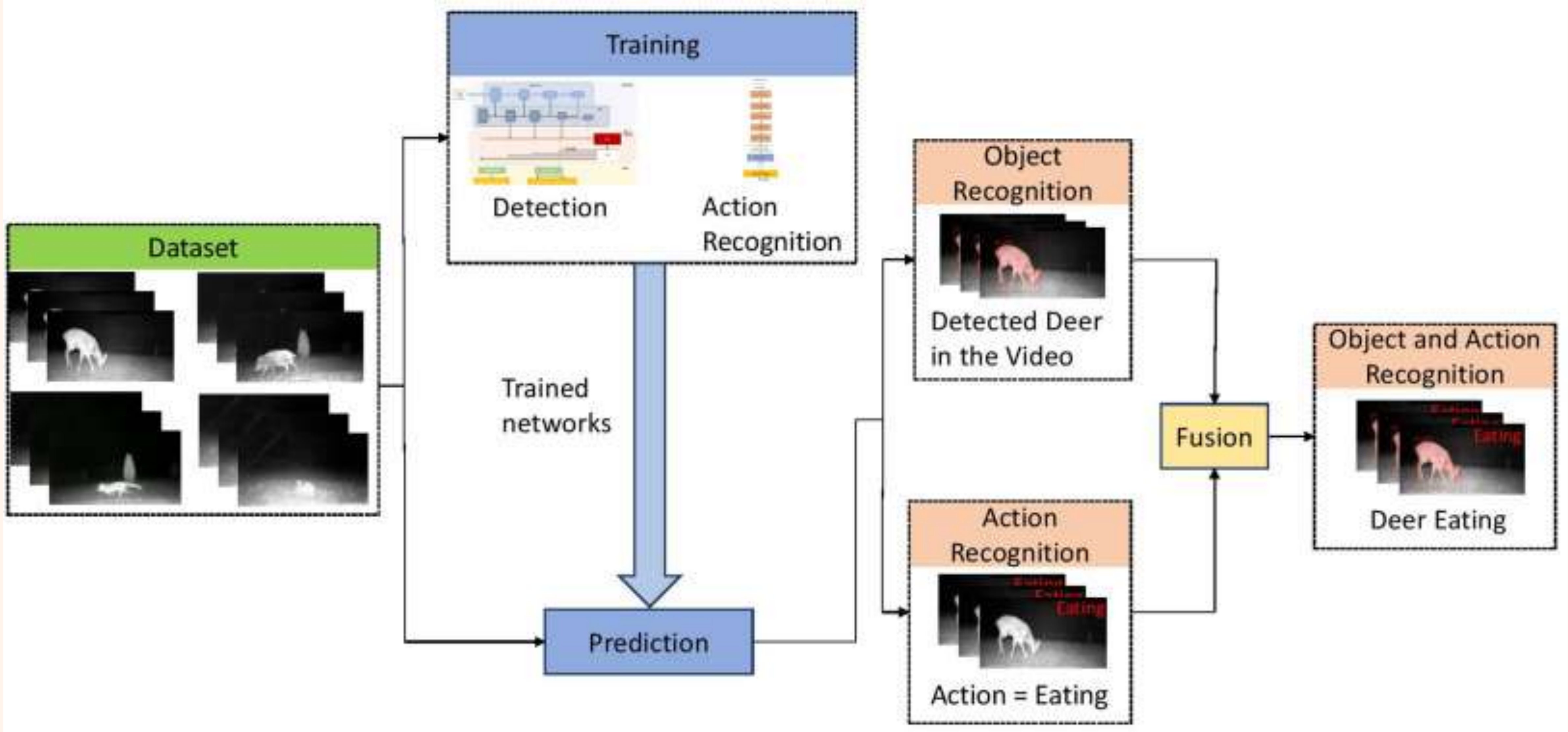
- We obtain data from the previously stated sources and are required to identify (multiple) animals clearly within the image via a bounding box



Problem Explanation

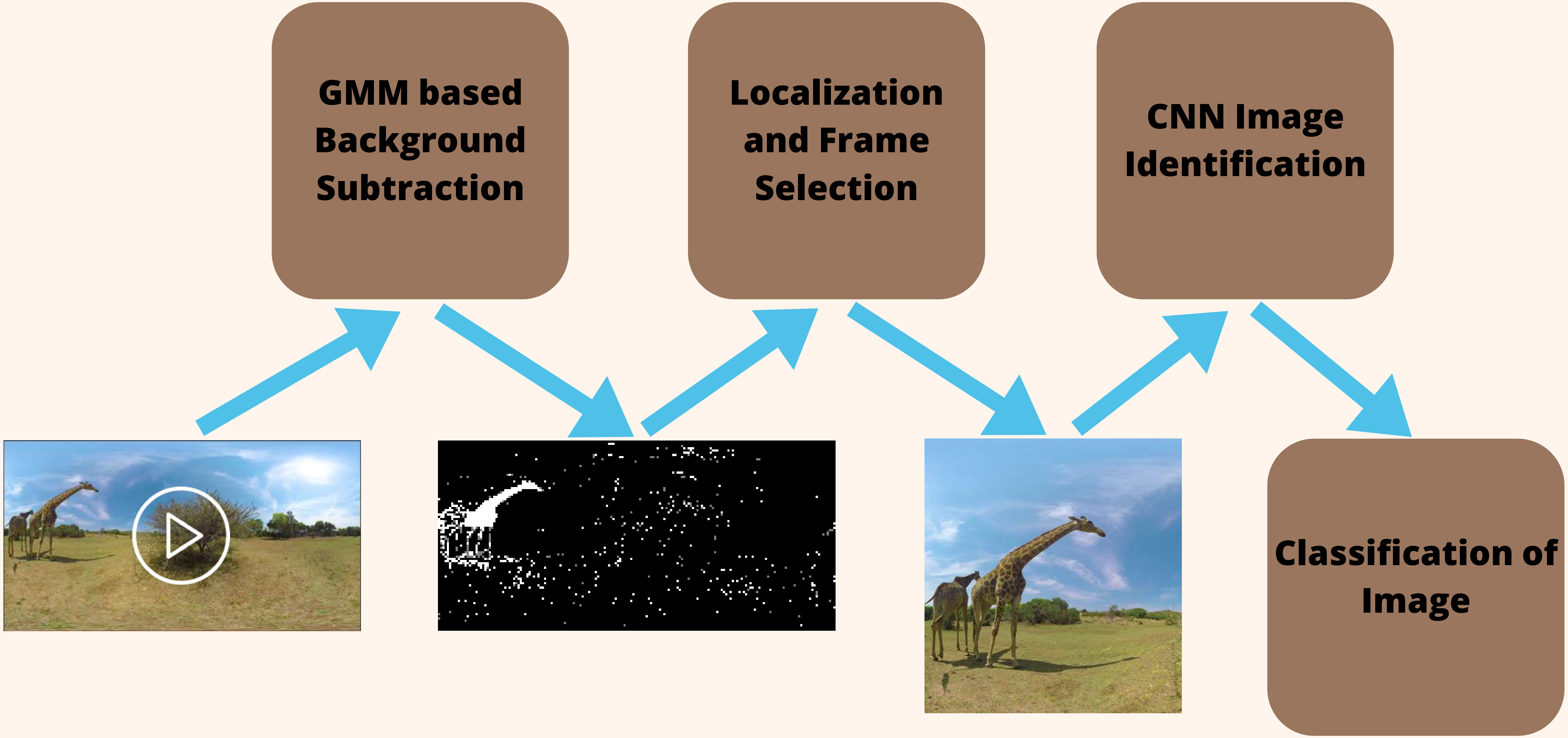
Level 2

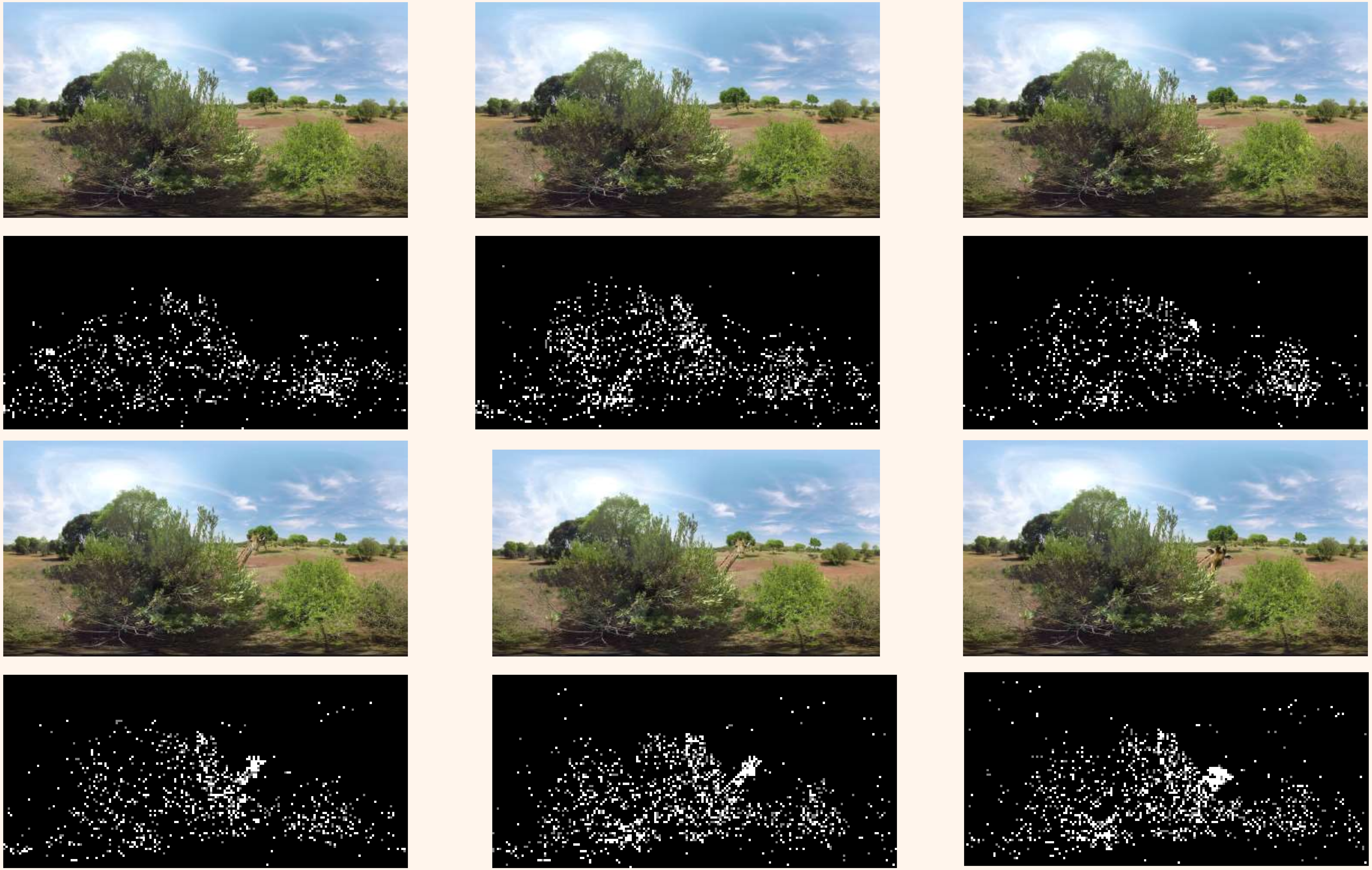
- In addition to level 1, we identify the action that an animal is engaging in

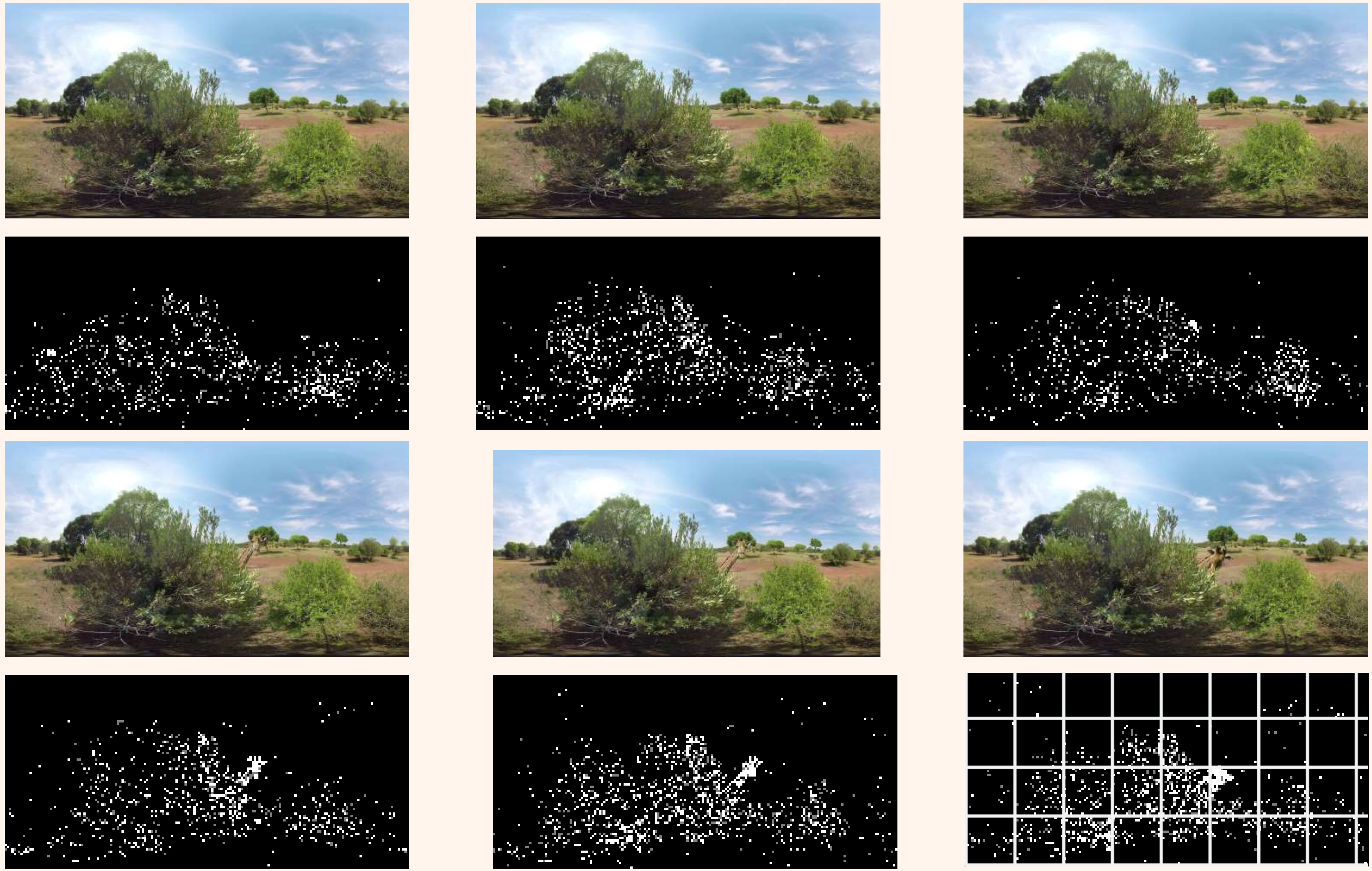


Video Feed

Pipeline


GMM based Background Subtraction
Localization and Frame Selection
CNN Image Identification
Classification of Image





Preprocessing

- We resize our frames to reduce the computational overhead for background subtraction.
- Taking every frame isn’t required, instead we empirically take every 10th frame to compute the background statistics.

Zivkovic, Zoran, and Ferdinand Van Der Heijden. "Efficient adaptive density estimation per image pixel for the task of background subtraction." Pattern recognition letters 27.7 (2006): 773-780.

GMM? What's that?

- Challenge: Identify background in a dynamically changing scene.
- Images of the scene without the intruding objects exhibit some regular behavior that can be well described by a statistical model.

Zivkovic, Zoran, and Ferdinand Van Der Heijden. "Efficient adaptive density estimation per image pixel for the task of background subtraction." Pattern recognition letters 27.7 (2006): 773-780.

GMM? What's that?

- Pixel-based background subtraction involves decision if the pixel belongs to the background (BG) or some foreground object (FG).

$$\frac{p(\mathrm{BG}|\vec{x}^{(t)})}{p(\mathrm{FG}|\vec{x}^{(t)})} = \frac{p(\vec{x}^{(t)}|\mathrm{BG})p(\mathrm{BG})}{p(\vec{x}^{(t)}|\mathrm{FG})p(\mathrm{FG})}$$

Zivkovic, Zoran, and Ferdinand Van Der Heijden. "Efficient adaptive density estimation per image pixel for the task of background subtraction." Pattern recognition letters 27.7 (2006): 773-780.

GMM? What's that?

- Pixel-based background subtraction involves decision if the pixel belongs to the background (BG) or some foreground object (FG).

$$\frac{p(\mathrm{BG}|\vec{x}^{(t)})}{p(\mathrm{FG}|\vec{x}^{(t)})} = \frac{p(\vec{x}^{(t)}|\mathrm{BG})p(\mathrm{BG})}{p(\vec{x}^{(t)}|\mathrm{FG})p(\mathrm{FG})}$$

$$p(\vec{x}^{(t)}|\mathrm{BG}) > c_{\mathrm{thr}}(= p(\vec{x}^{(t)}|\mathrm{FG})p(\mathrm{FG})/p(\mathrm{BG}))$$

  - How do we estimate this density online?

Zivkovic, Zoran, and Ferdinand Van Der Heijden. "Efficient adaptive density estimation per image pixel for the task of background subtraction." Pattern recognition letters 27.7 (2006): 773-780.

GMM? What’s that?

- Decision based on training data, for each new sample we re-estimate density
- Modelled as GMM with M components

$$\hat{p}(\vec{x}|\mathcal{X}_T, \mathrm{BG}+\mathrm{FG}) = \sum_{m=1}^{M} \hat{\pi}_m \mathcal{N}(\vec{x}; \hat{\vec{\mu}}_m, \hat{\sigma}_m^2 I)$$

Zivkovic, Zoran, and Ferdinand Van Der Heijden. "Efficient adaptive density estimation per image pixel for the task of background subtraction." Pattern recognition letters 27.7 (2006): 773-780.

GMM? What's that?

- We update the mixing weights, means and variance using:

$$\hat{\pi}_m \leftarrow \hat{\pi}_m + \alpha(o_m^{(t)} - \hat{\pi}_m),$$
$$\hat{\vec{\mu}}_m \leftarrow \hat{\vec{\mu}}_m + o_m^{(t)}(\alpha/\hat{\pi}_m)\vec{\delta}_m,$$
$$\hat{\sigma}_m^2 \leftarrow \hat{\sigma}_m^2 + o_m^{(t)}(\alpha/\hat{\pi}_m)(\vec{\delta}_m^T\vec{\delta}_m - \hat{\sigma}_m^2)$$

- Obtained from Maximum Likelihood function of ownership
- Constant alpha defines exponentially decaying envelope; limits influence of old data

Zivkovic, Zoran, and Ferdinand Van Der Heijden. "Efficient adaptive density estimation per image pixel for the task of background subtraction." Pattern recognition letters 27.7 (2006): 773-780.

GMM? What's that?

- We use Mahalanobis distance to measure how far a point is from a distribution
- With this measure, check whether any component of our current model can "sufficiently represent" the current pixel value.
- If not, we introduce another Gaussian component with mean as the current pixel value

Zivkovic, Zoran, and Ferdinand Van Der Heijden. "Efficient adaptive density estimation per image pixel for the task of background subtraction." Pattern recognition letters 27.7 (2006): 773-780.

GMM? What's that?

- The background is mainly described by the prominent components of GMM

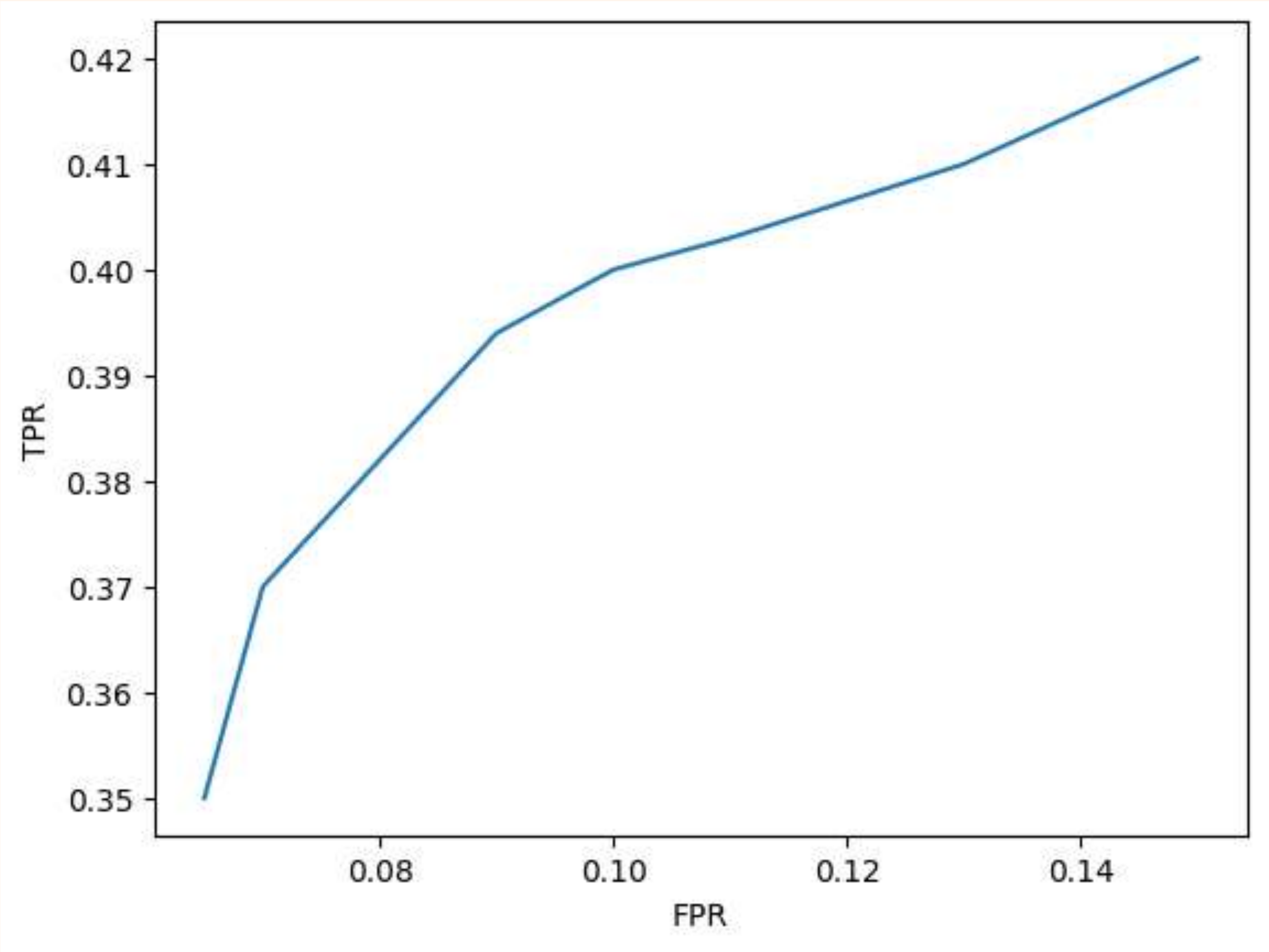
$$\hat{p}(\vec{x}|\mathcal{X}_T, \mathrm{BG}) \sim \sum_{m=1}^{B} \hat{\pi}_m \mathcal{N}(\vec{x}; \hat{\vec{\mu}}_m, \sigma_m^2 I).$$

$$B = \arg\min_{b} \left(\sum_{m=1}^{b} \hat{\pi}_m > (1 - c_{\mathrm{f}}) \right)$$

- If we cannot describe a pixel with these components, we deem it a foreground pixel

Zivkovic, Zoran, and Ferdinand Van Der Heijden. "Efficient adaptive density estimation per image pixel for the task of background subtraction." Pattern recognition letters 27.7 (2006): 773-780.

ROC for Detection

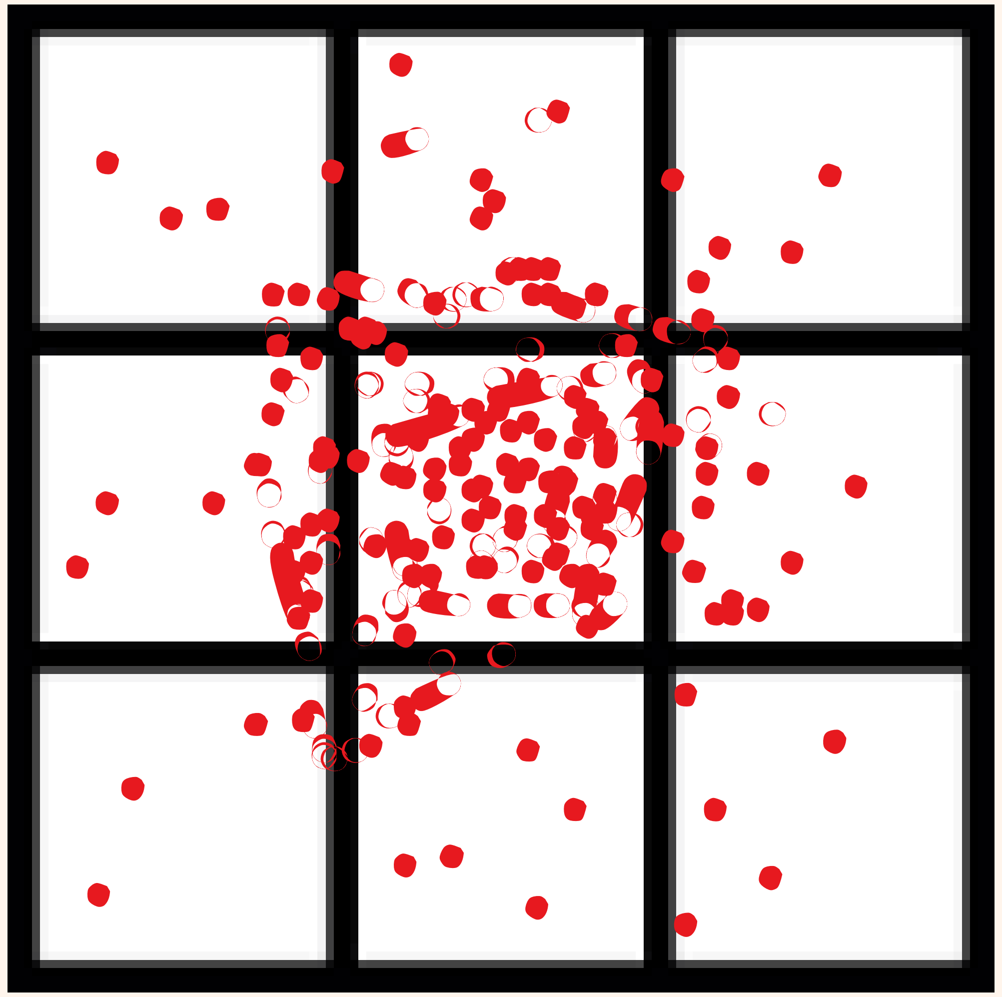


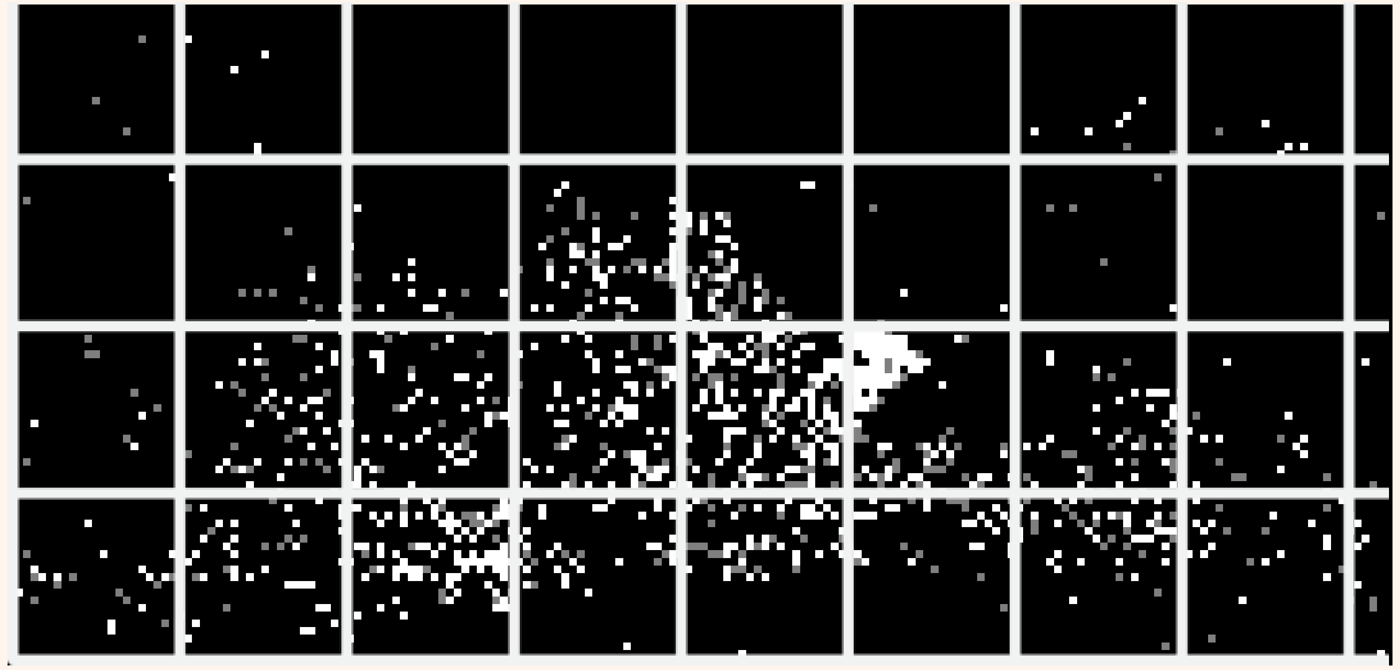
Zivkovic, Zoran, and Ferdinand Van Der Heijden. "Efficient adaptive density estimation per image pixel for the task of background subtraction." Pattern recognition letters 27.7 (2006): 773-780.

Cropping the selected frame

- After we have estimated the foreground masked image we divide the image into a 4X8 grid.
- We find the square with the highest foreground density compared to the background.
- To ensure that we don't miss out on the part which contains the animal we also consider the squares surrounding the square with maximum foreground pixel density

Select and Crop

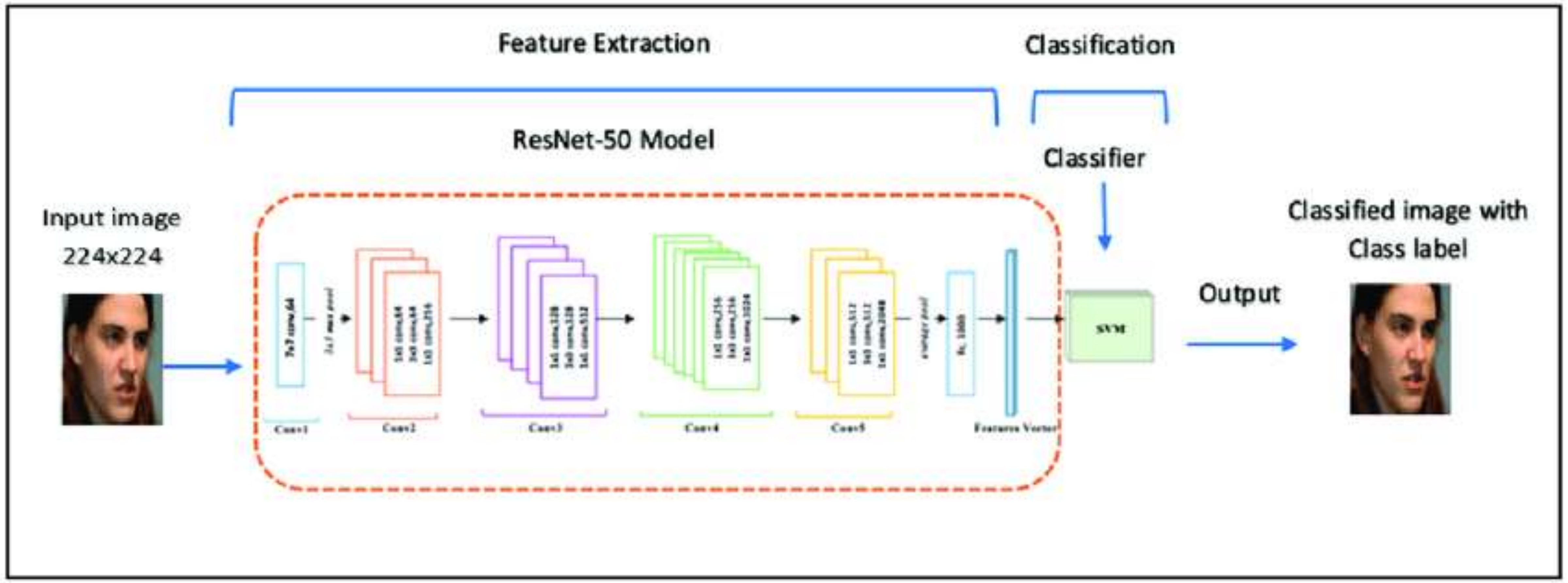




Dropping redundant images

- For efficient use of resources, we will try not to process redundant images, thus we choose only the images which satisfy certain conditions
- For each image foreground mask, we calculate the square with the largest pixel density, we will take this image only if this exceeds 1.1*(previous frame max density).

ResNet CNN



Feature Extraction
Classification
ResNet-50 Model
Classifier
Input image
224x224
Classified image with
Class label
Output
SVM

ResNet CNN training

- We Trained CNN using ikomia API, we handpicked quality images from the Serengeti Snapshot dataset of giraffe, zebra, and wildebeest and trained our CNN on these images, and built a classifier.

- Our CNN can identify whether an animal is present in the image or not, if yes, it does the classification with an accuracy of 90 per cent.

Future Scope

- Using K-Means clustering to obtain multiple animal images from one frame.

THANK YOU!