Classical Computer Vision Still has its Place in Object Detection

Nick Cantalupa and Sean Duffy

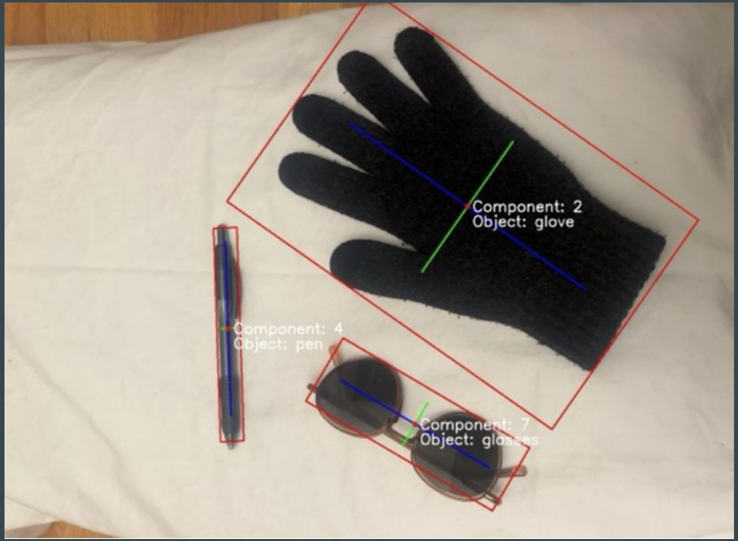
Previous Work

Features:

- Can accurately detect objects at different orientations and scales
- Can learn new objects
- Uses 9 manually calculated features for matching

Limitations:

- Requires good lighting and white background
- Slow (<1 FPS)
- Some errors in matching
- Detects objects where there are none

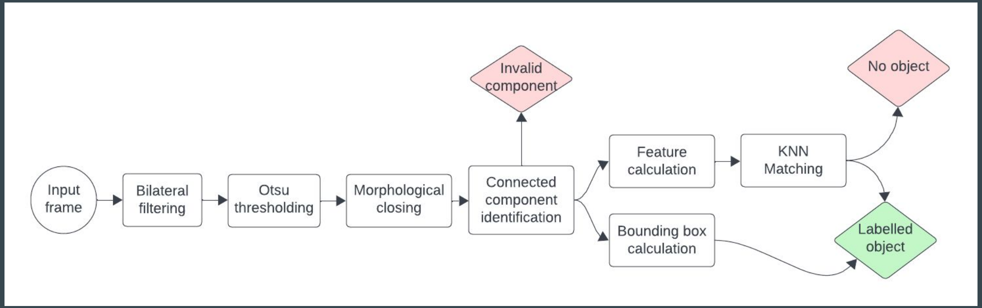


Goals

- Improve classical object detection system
- Implement deep learning based object detection system
- Evaluate and compare the performance of the two systems
- Evaluate the response of the systems to adversarial attacks

Classical Detection Improvements

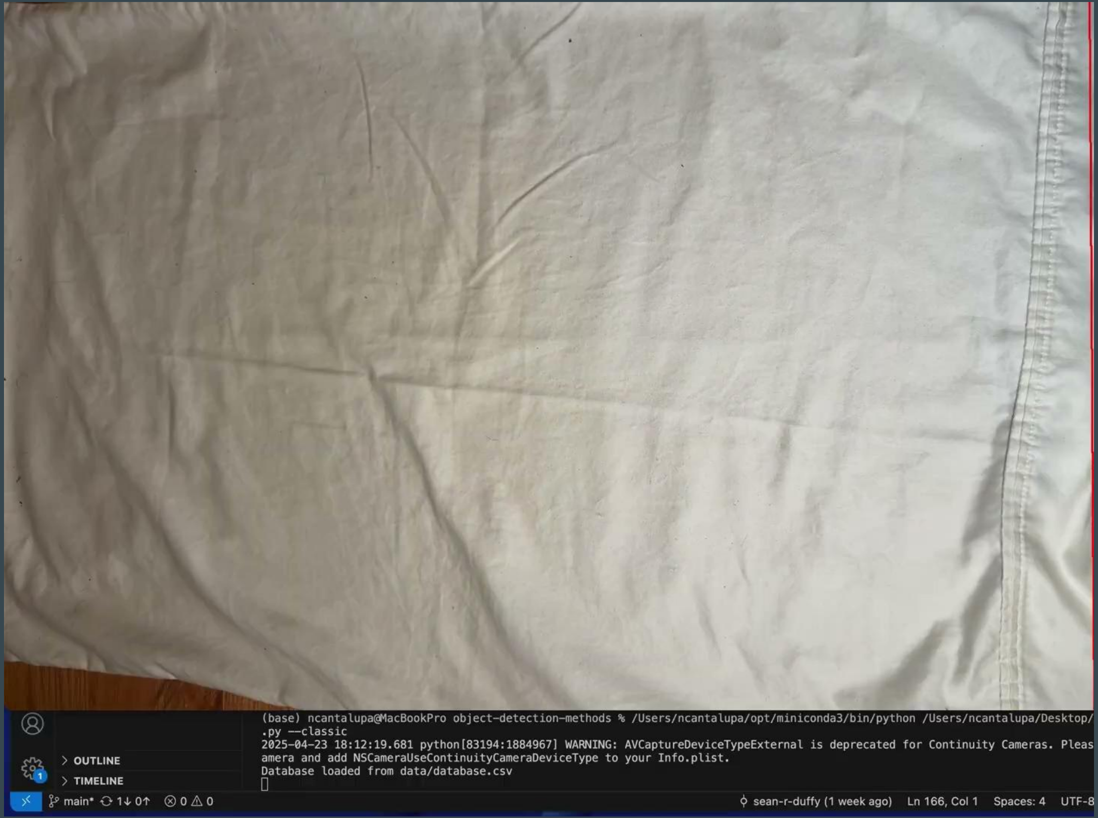
- Re-implemented in Python for easy integration with deep learning methods
- Thresholding and morphological filtering improved
- Component thresholding introduced
- Improved database handling
  - Normalized database stored in memory
  - Distance threshold (components not identified as objects unless they are nearby in feature space)
  - Easier training (objects can be labelled quickly and easily)
- Higher FPS



Deep Learning

- Implemented 2 transformer models from Hugging Face
- **facebook/DETR-ResNet-50**
  - Larger model: 41.6M parameters
  - Trained on COCO 2017 image dataset with 80 classes
  - Released by Carion et al. from Meta in 2020
- **hustvl/yolos-tiny**
  - Smaller model: 6.5M parameters
  - Trained on ImageNet-1k, then fine-tuned on COCO 2017
  - Released by Fang et al. from

Demo


OUTLINE
TIMELINE
(base) ncantalupa@MacBookPro object-detection-methods % /Users/ncantalupa/opt/miniconda3/bin/python /Users/ncantalupa/Desktop/
.py --classic
2025-04-23 18:12:19.681 python[83194:1884967] WARNING: AVCaptureDeviceTypeExternal is deprecated for Continuity Cameras. Pleas
amera and add NSCameraUseContinuityCameraDeviceType to your Info.plist.
Database loaded from data/database.csv
main* 1↓ 0↑ 0 0
sean-r-duffy (1 week ago)
Ln 166, Col 1
Spaces: 4
UTF-8

Adversarial Attacks

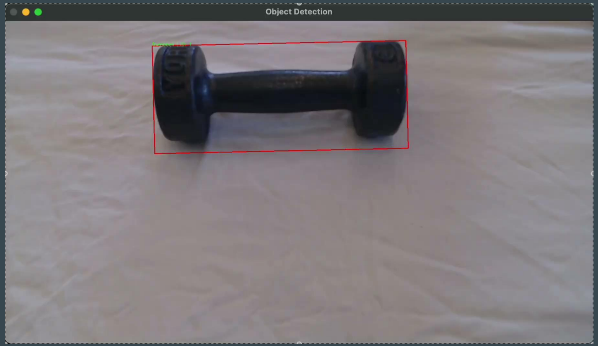
- Patches were calculated as in Brown et al.
- Sample images were collected that resemble the testing environment.
- The patch was added with variation in scale, location, and orientation
- Initialize with random noise, the patch is updated by the gradients with respect to the loss function against the target class
- Target classes were remote and car
    - Both classes in ImageNet and COCO
- Can be overlaid both physically and digitally
- The patches were ineffective against the deep learning methods, and only partially effective against the classical method

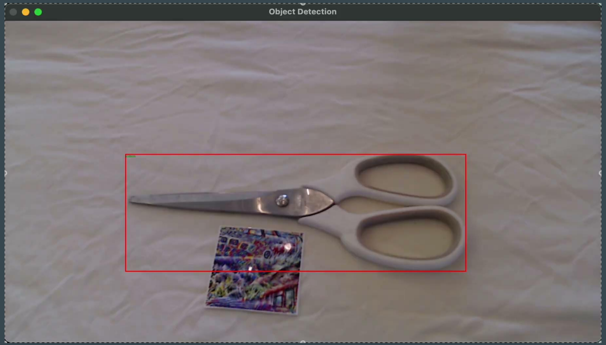
Adversarial Car

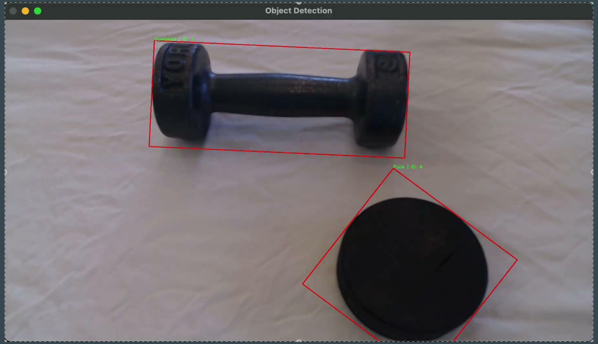
Adversarial Remote

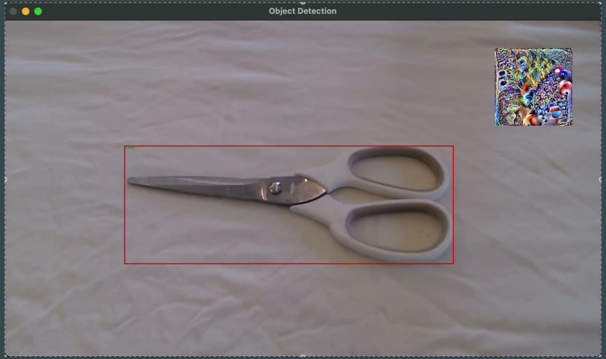
Training example

Adversarial Attacks - Demo









Conclusions

- Classical CV Object Detection
  - Faster frame rates for real time applications
  - Easy to add new classes, training data
  - Worse at generalizing, more situational
  - Struggles with reflective surfaces
- Deep Networks
  - High accuracy, works in many situations
  - Stable in different lighting
  - Slower for real time processing
  - Requires retraining to add new classes
- Adversarial Attacks
  - Ineffective against deep learning methods
  - No more effective than another object