



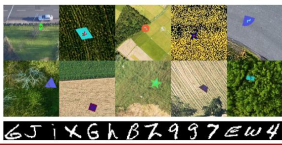
# Standardized Object Detection and Classification for Unmanned Aerial Vehicles

Joshua F. Payne, Peggy (Yuchun) Wang  
{joshp007, wangyuc}@stanford.edu



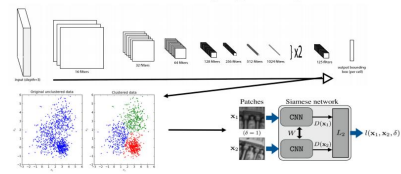
## Problem Statement

- Goal:** Detect, localize, and classify the shape, color, and alphanumeric character of a poster object from an aerial image
- Datasets:** (1) Extended-MNIST, and (2) created RGB dataset by placing generated geometric shapes with alphanumeric characters onto scraped aerial views of fields. Generated parallel XML files denoting bounding regions.



## Approach

We used the YOLO algorithm model to localize object and classify shapes, K-means clustering for segmenting the image and isolating the alphanumeric, and we used both a convolutional neural network and Siamese convolutional neural network for classifying the alphanumeric.

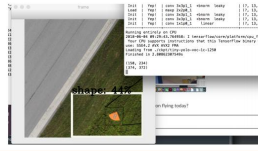


## Acknowledgements

We are grateful to Ahmadrza Momeni and the rest of the CS 230 teaching staff for their support.

## (1) YOLO (You Only Look Once) Network

- Based on Darkflow's Tiny-YOLO model
- Processes 1080x1920 RGB images on a 16GB CPU at ~4 FPS
- Performed well with detection/localization

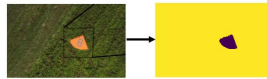


## (2) Segmentation

- Segmented the image using k-means clustering (2 clusters):
- Used Euclidean norm to calculate nearest template color to average color of shape:

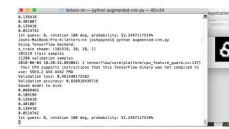
$$\sum_{j=1}^n \|x_i(j) - c_j\|^2$$

$$\sum_{i=1}^3 \|x_i - t_i\|^2$$



## (3) Convolutional Neural Network

- Used 1 convolutional and pooling layers, 2 dense hidden layers
- Augmented data in-training
- Performed even better with real data because of EMNIST Bayes
- Used learning reduction on plateau, dropout
- Used cross-entropy loss function:



$$-\sum_{c=1}^{47} y_{o,c} \log(p_{o,c})$$

## (4) Siamese Convolutional Neural Network

- Used positive/negative pairings to learn encodings for alphanumeric images
- Same layers as (3)
- These can be visualized using t-SNE →
- Used contrastive loss function:

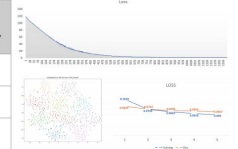
$$(1 - Y) \frac{1}{2} (D_w)^2 + Y \frac{1}{2} \{ \max(0, \alpha - D_w) \}^2$$



## Results and Discussion

- |  |   |
|--|---|
| <p><b>Detection</b></p> <ul style="list-style-type: none"> <li>Training set: 10,000 images, Dev set: 1,000</li> <li>Detection accuracy is good, classification accuracy is poor due to loss function</li> <li>Loss convergence, training speed didn't change with addition of classes</li> </ul> | <p><b>Alphanumeric</b></p> <ul style="list-style-type: none"> <li>Training set: 107,159 images, Test set: 5,640 - CNN</li> <li>Training set: 200,000 pairs, Test set: 10,000 pairs - S-CNN</li> <li>Siamese CNN has better accuracy than CNN due to learning encodings</li> </ul> |
|--|---|

Model	Training Accuracy	Dev Accuracy
YOLO	92.30%	91.80%
CNN	86.99%	84.65%
Siamese	97.68%	97.08%



## Future Work

- Explore using Siamese CNN model for use in alphanumeric character classification
- Implement a separate neural network for classifying the shape, because sometimes YOLO confuses certain shapes with others even if it correctly guesses the bounding boxes for shapes
- Tackle tougher problems like search-and-rescue operation detection, infrastructure assessment using 3-D internal models and capsule networks

## References

[1] Abadi, Martin et al. "TensorFlow: A System for Large-Scale Machine Learning." OSDI, Vol. 16, 2016.  
 [2] AUCU-SIAD: 2018 Competition Rules. [http://www.aucsiad.com/organization/competitions/2018/aucsiad\\_2018\\_rules.pdf](http://www.aucsiad.com/organization/competitions/2018/aucsiad_2018_rules.pdf).  
 [3] Bai, Heideft, Timm Tophelaars, and Luc Van Gool. "Surf: Speeded up robust features." Neural Networks (IJCNN), 2017 International Joint Conference on. IEEE, 2017.  
 [4] Cohen, Gregory, et al. "EMNIST: Extending MNIST to handwritten letters." Neural Networks (IJCNN), 2017 International Joint Conference on. IEEE, 2017.  
 [5] Darkflow: a tensorflow implementation for YOLO. <https://github.com/thtrieu/darkflow>.  
 [6] Hartigan, John A., and Manchek A. Wong. "Algorithm AS 136: A k-means clustering algorithm." Journal of the Royal Statistical Society, Series C (Applied Statistics) 26, 1 (1977): 100-108.  
 [7] Keras: The Python Deep Learning Library.  
 [8] Koch, Gregory, Richard Ganeff, and Rustan Salakhutdinov. "Siamese neural networks for one-shot image recognition." ICM: Deep Learning Workshop, Vol. 2, 2015.  
 [9] Maten, Laurent van der, and Geoffrey Hinton. "Visualizing data using t-SNE." Journal of machine learning research 9, Nov (2008): 2579-2906.  
 [10] OpenCV: the Open Source Computer Vision Library.  
 [11] Pandas: the Python Data Analysis Library.  
 [12] Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." Journal of machine learning research 12, Dec (2011): 2825-2830.  
 [13] PIL: Python Imaging Library.  
 [14] Redmon, Joseph, and Ali Farhadi. "YOLO9000: better, faster, stronger. CoRR abs/1612.08424 (2016)." (2016).  
 [15] Siamese 2-D encoding visualization implementation. [https://github.com/yekwo/siamese\\_if\\_mnist](https://github.com/yekwo/siamese_if_mnist)