



# Vision: Accurate Color Extraction from Photographs

Devansh Desai | Shamika Kulkarni | Shweta Patil | Nate Wilson

# The Challenge:

Background:

Vision is a retail start-up founded in Charlotte, NC.

Vision's app enables color analysis and search of retail products, with currently 3000 colors available for search and find complementary products.

Our Objective:

Given an image, we want to identify the true color of the foreground object.

Our Approach:

Images are assumed to be from a phone camera and not downloaded.

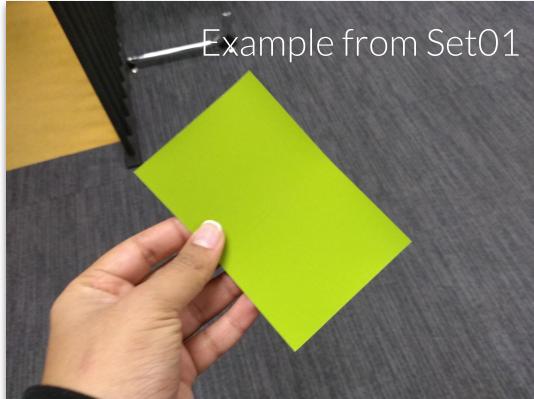
The center pixel must have the object whose true color needs to be identified.

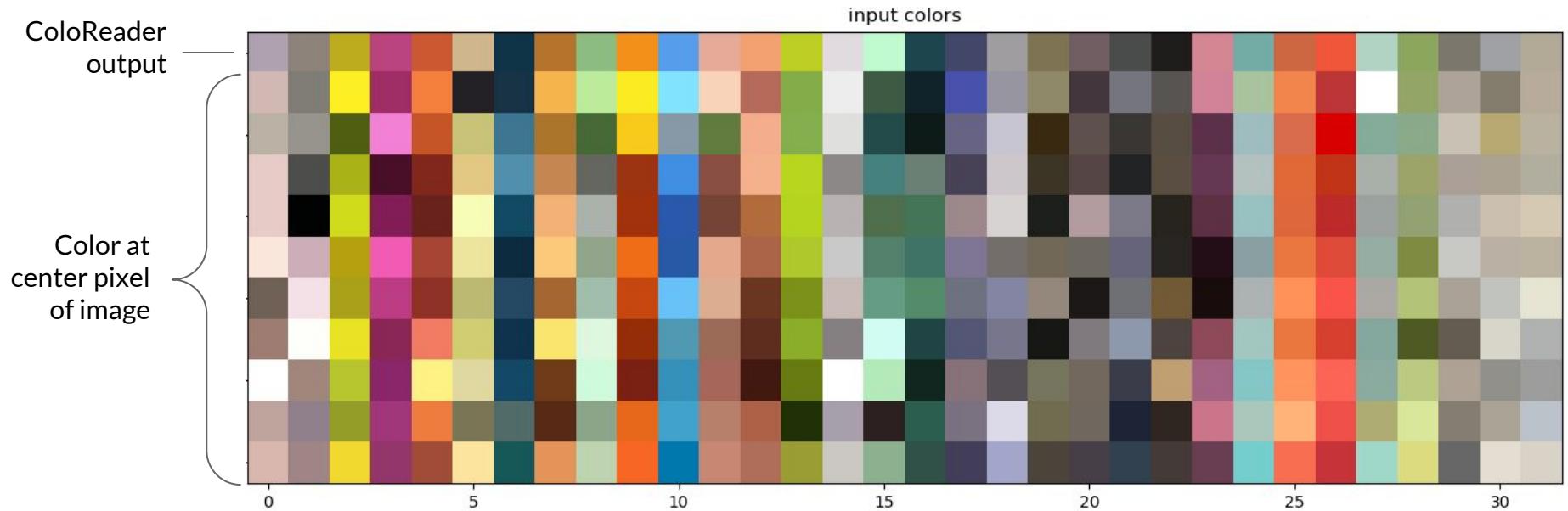
The object in the image needs to be mono-color.



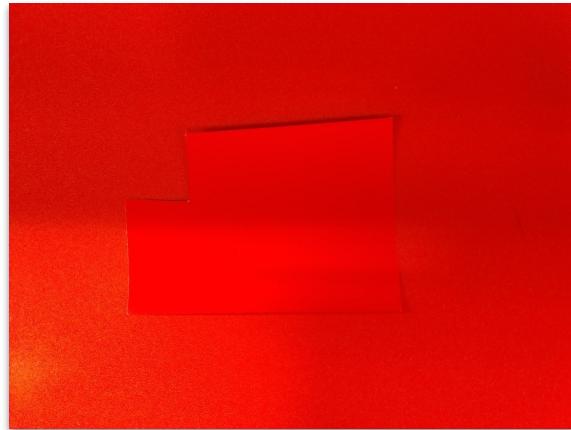
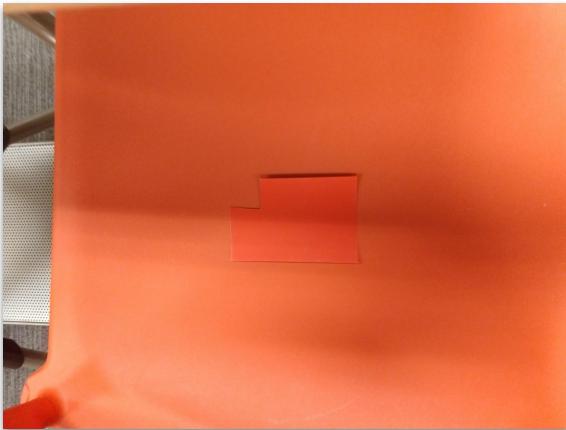
# The Data:

- ❖ Collected dataset by taking pictures of 32 different color cards under 10 different lighting or shadow conditions resulting in 320 images
- ❖ Second set of 12 colors under 2 different lighting conditions resulting in 24 images for regression analysis
- ❖ Metadata including ISO and Exposure Time



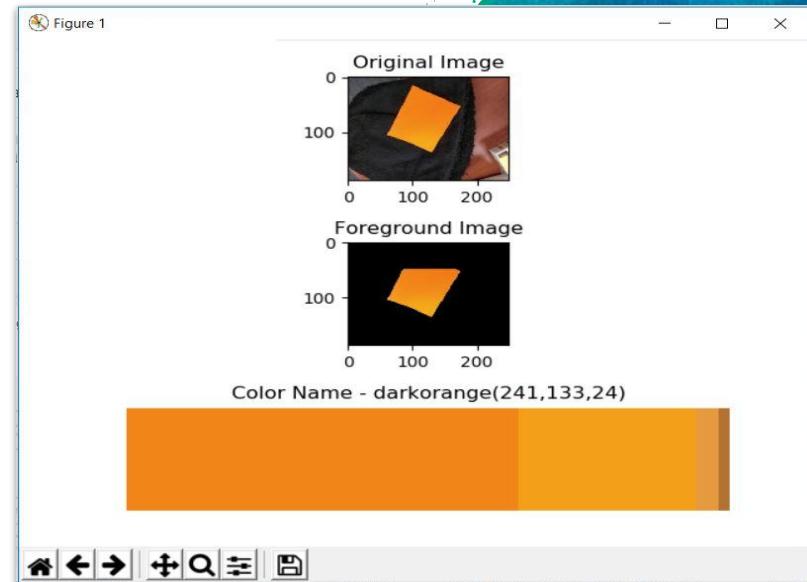


These are the same color!



# The Data Input:

- **Center-mode:** center pixel, metadata
- **Metrics-mode:** center pixel, picture mean, max, min, variance, and standard deviation, metadata
- **Slice-mode:** subset of image pixels, metadata
- **Full-mode:** all pixel, metadata
- **K means:** segmentation color clustering, metadata
- **RGB and HLS**



# The Algorithm Structure:

Classification:

- Treating it as a classification problem using softmax and cross entropy loss
- 32 different classification, simply looking for correlation between inputs
- Many different settings for learning rate, data split, batch size and network structure depth and breadth were used

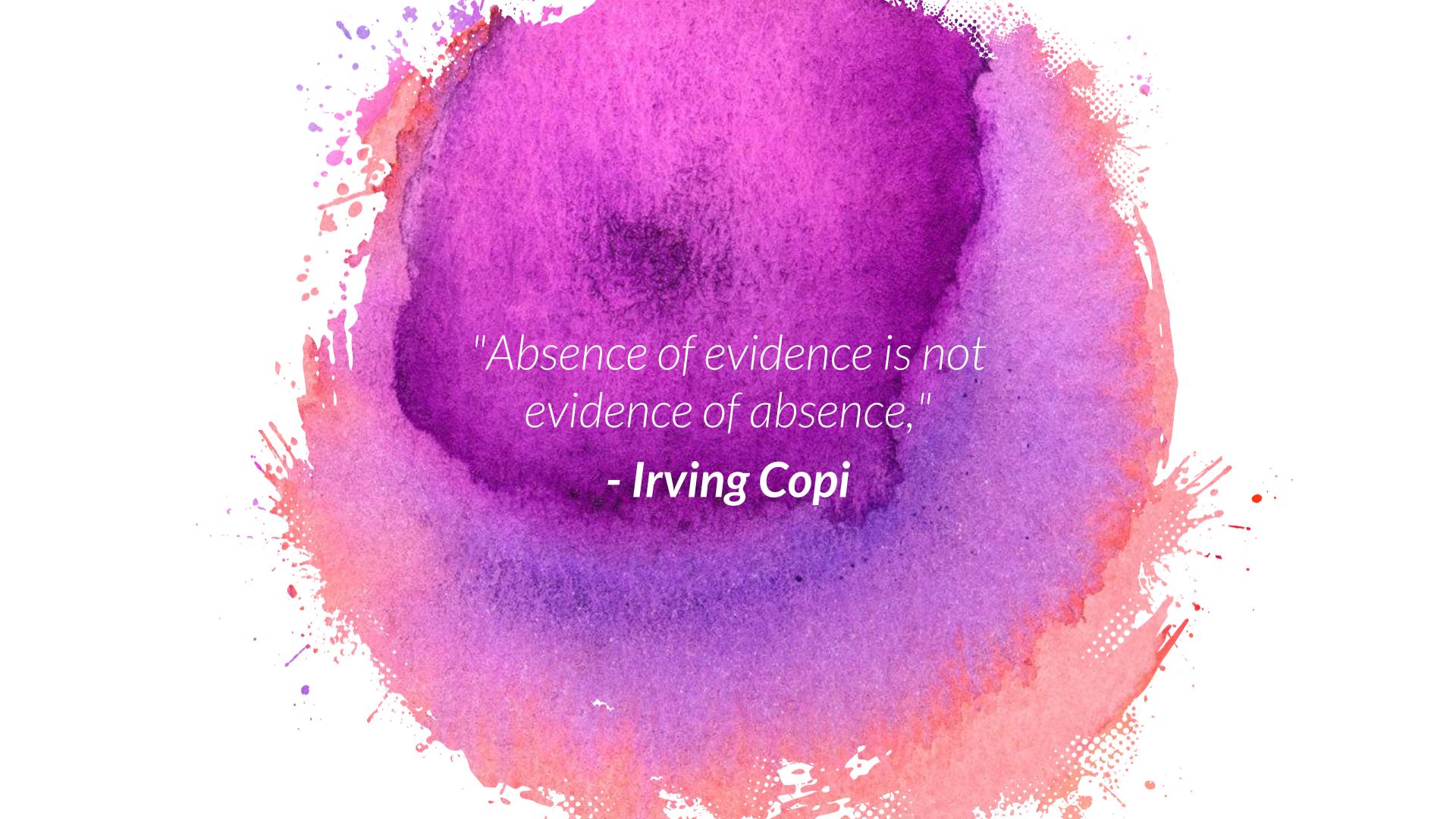
Regression:

- Treating it as a regression problem using mean square error and mean average error loss
- Mapping the color space to find a linear relationship between inputs
- Similar variance in structure
- Most successful structure consisted mirrored the following equation:  $(iso^*w1+w2)^*[H,S,L]$



# The Results:



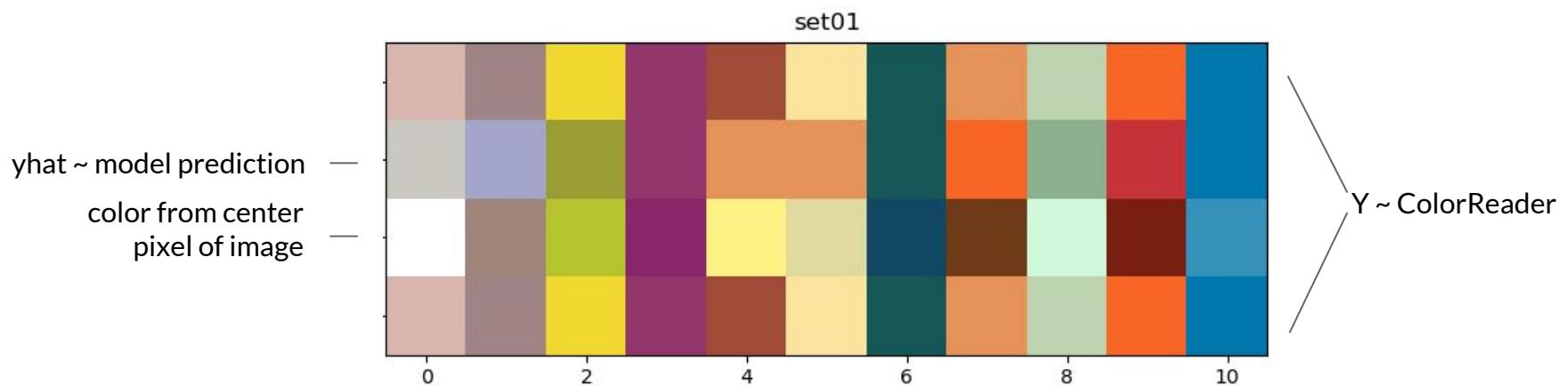
The background of the image is a white surface with a large, irregular ink splatter in shades of purple and red. The ink has spread out, creating a textured, watercolor-like effect with visible brushstrokes and splatters of varying sizes.

*"Absence of evidence is not  
evidence of absence,"*

*- Irving Copi*

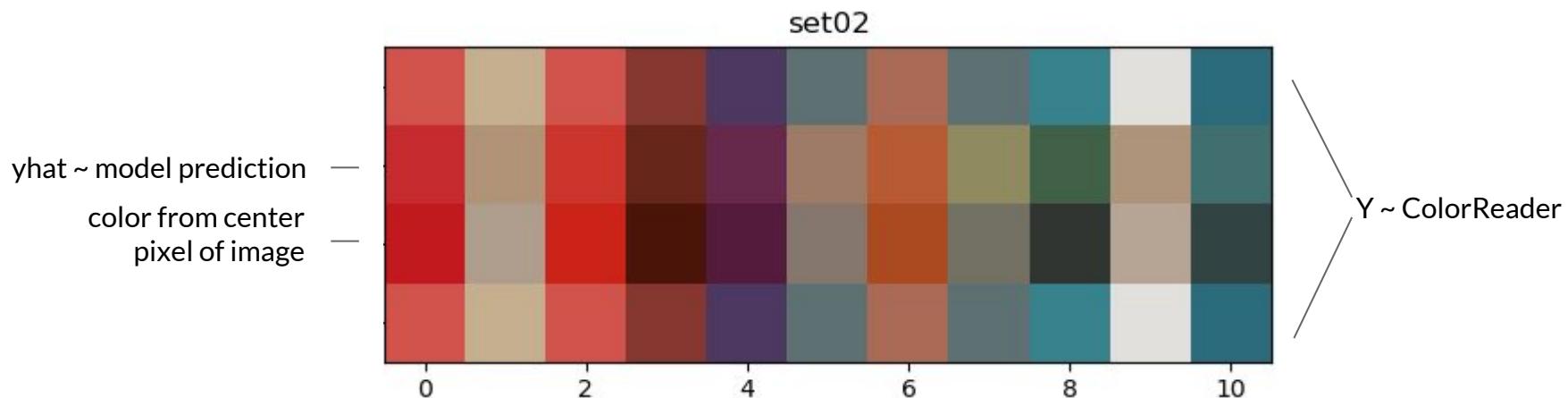
# Classification:

Model Accuracy: 40%



# Regression:

Color from center pixel compared to  $y$  [mse]: 0.0200729  
 $\hat{y}$  ~model prediction [mse]: 0.0157040



# The Realization:

1. Limited dataset with low camera control
2. Lack of correlation between photos
3. Finding correct parameters for Neural Network training is very hard



# The Possible Solution:

Because Vishion's application is iOS or eventually Android-based, they can access the camera controls directly, which would allow for finer adjustments and maintain a uniformity throughout the pictures.

Metrics that might be useful:

- Forcing the user to use a gray card (or a piece of paper)
- Doing an exposure sweep to calibrate like an HDR photo
- Forcing the camera to use flash



A SPECIAL THANK YOU TO  
VISHION AND PROFESSOR WELCH FOR  
ESTABLISHING THIS PARTNERSHIP!