Cats & Dogs

By Group 7, Xiaoyu Wang, Yicheng Wang, Ouwen Sha, Husam Abdul-Kafi, Yimin Zhang

How are dogs and cats different in computer’s eye?

Our experiment

Features

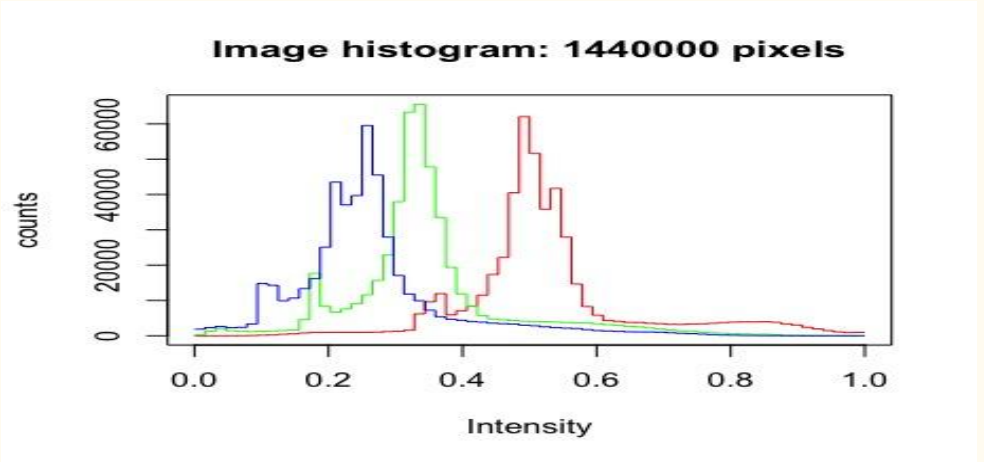
Everything including but not limited to: color, outline, shape

- Color (RBG+HSV)
- Momocs (outline + contour)
- Laplacian filter
- OpenCV, etc

Color

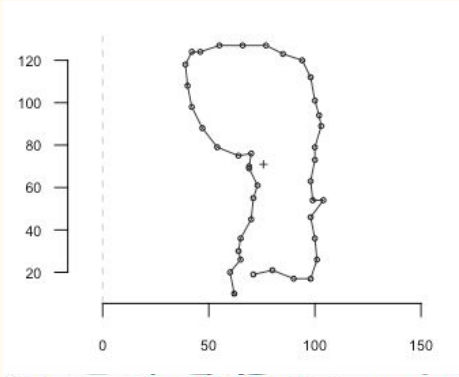
1. Served as our baseline model
2. Subdivide pixel values in each color channels into multiple bands. The color feature from RGB describes the color distribution of the image
3. Hue, Saturation, Value (HSV) is easier to interpret than RGB as it is closer to human perception

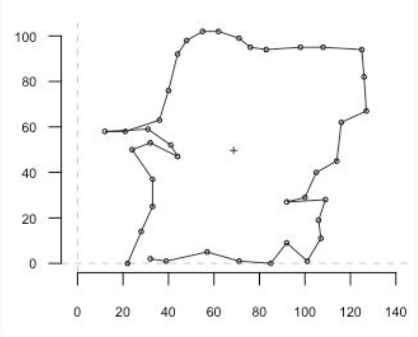
```
> print(error.rate)
[1] 0.3353741
> print(best.error.rate)
[1] 0.3326531
>
```



Momocs

1. Momocs extracts contour from the gray scale laplacian filter
2. It then gathers coordinate information from the contour
3. Momocs then find coordinate coefficient from the coordinate information






```
feature <- function(img_dir,img_name,data_name = NULL){

  n <- length(img_name)
  #n <- 50
  dat <- list()
  for(i in 1:n){
    if(trunc(i/5)*5==i){print(i)}

    #read images
    img <- readImage(paste0(img_dir,img_name)[i]) #;display(mat)
    img <- channel(img, mode="gray") #;display(img)

    #High-pass Laplacian filtering
    f_high <- matrix(1, nc=3, nr=3)
    f_high[2,2] <- -8
    img <- filter2(img, f_high)

    #Adaptive thresholding
    img <- thresh(img, w = 50, h = 50, offset = 0.05) #;display(final_img)
    img <- resize(img, 128,128)
    oc <- ocontour(bwlabel(img)) #;oc
    max <- c()
    for(j in 1:length(oc)){
     max[j] <- nrow(oc[[j]])
    }
    #max
    #plot(oc[[which.max(max)]], type='l')
    coo <- coo_sample(oc[[which.max(max)]], 40) #;coo
    coo %>% coo_plot(points=TRUE)
    dat[[i]] <- coo
  }
  return(dat)
```

OpenCV(experiment)

1. Determines corner by determining the average change of intensity from shifting small window

2. From these points, correspondences between images are drawn

Final Model

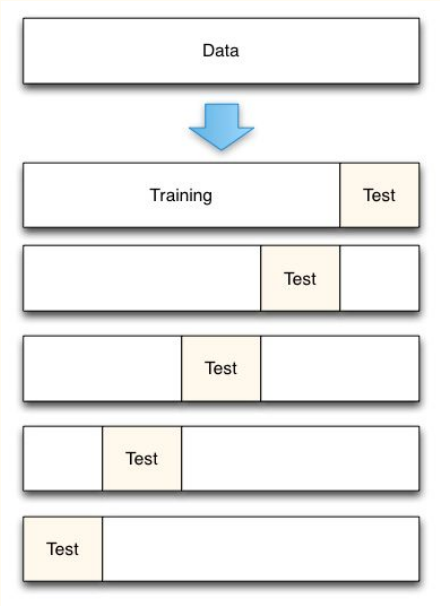
SIFT

Scale Invariant Feature Transformation

1. Scale Space Extrema Detection
2. Refine to get more accurate results
3. Assign orientation to each key point to achieve invariance to image rotation

Cross Validation

1. Avoid over-fitting
2. Holding out data to test



Thank you