

Optical Character Recognition and Correction for Images of Digital Displays

Brian Cornet

May 3rd, 2021

DSC 498 – Dr. Donghui Yan

DSC 499 – Dr. Scott Field

Part 1: What's the deal?

Purpose

- A platform for **collecting data from images**
 - Data is not private, but collection by hand is tedious and prone to error



Final Fantasy Tactics (Sony PlayStation)



ID	Name	Job	CHP	MHP	CMP	MMP	...	Acc	Set1	Set2	React	Support	Move	
7	Agrias	Holy Knight	559	559	81	81	...		Bracer	Holy Sword	White Magic	Auto Potion	Attack UP	Teleport

[1 rows x 35 columns]

This took me 3 minutes to type!
I need to do it 20 more times...

Surely there's a better way?

Optical Character Recognition (OCR)

- Converts images of text into digital text
- Commonly used for scanning documents and in AI services (self-driving cars, etc.)
- Typically relies on pattern recognition



Optical Character Recognition (OCR)

- Relies on predictive modeling from machine learning
- If a human can't read it, chances are a machine can't either!

② Chord, secant and Newton's methods

• Idea: use more information than just the sign of the function f

e.g. $f(x^{(k)})$, $f'(x^{(k)})$...

Review Taylor Series Expansion:

→ approximating a function $f(x)$ "around" some fixed point x

← (Thanks Yanlai!)

Line acc:

73.23%

85.00%

40.00%

86.67%

86.49%

OCR Interpretation (74.28% Mean Accuracy)

Chord secant and Navon's adeterods

Idea use more information than just the sign of the function f

ego fexch f exam

Review Taylor Serves Expansive

approximating a fraction fun around some fixed point x

Actual Text

② Chord, Secant, and Newton's methods

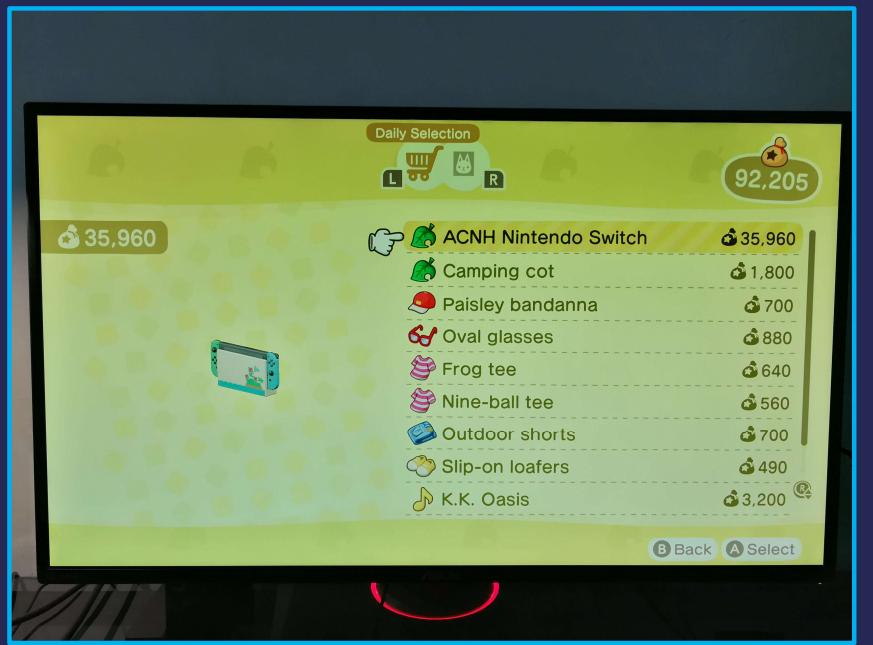
• Idea: use more information than just the sign of the function f
e.g. $f(x^{(k)})$, $f'(x^{(k)})$...

Review Taylor Series Expansion:

→ approximating a function $f(x)$ "around" some fixed point x

Objectives

- **Create** platform for data collection from images (including images of images!)



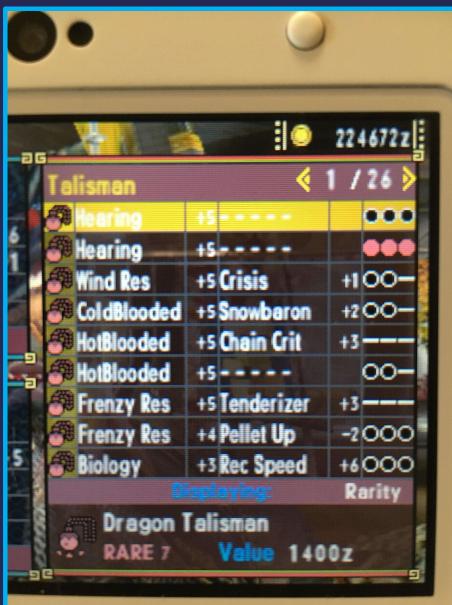
Animal Crossing: New Horizons (Nintendo Switch)

ACNH Nintendo Switch
Camping cot
Paisley bandanna
Oval glasses
Frog tee
Nine-ball tee
Outdoor shorts
Slip-on loafers
K.K. Oasis

Lists!

Objectives

- **Create** platform for data collection from images (including images of images!)



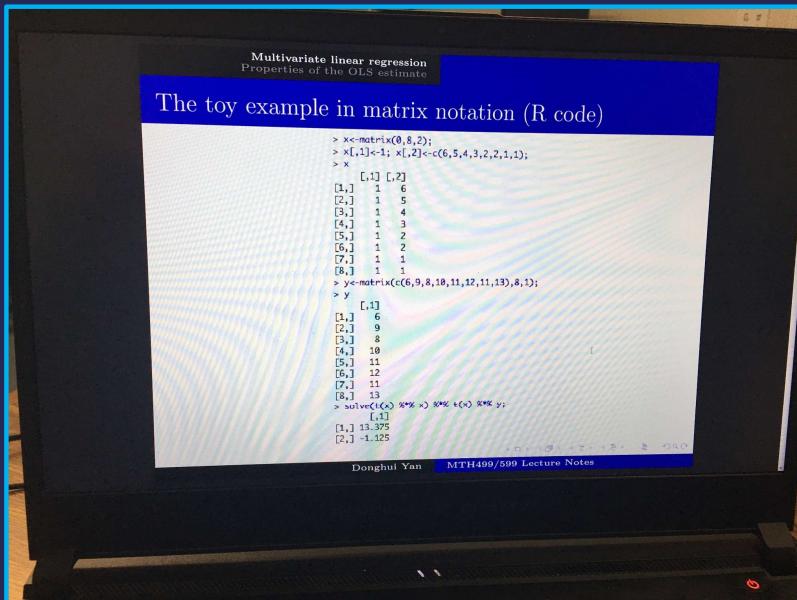
Hearing	+5	- - - - -		000
Hearing	+5	- - - - -		000
Wind Res	+5 Crisis	+100		
ColdBlooded	+5 Snowbaron	+200		
HotBlooded	+5 Chain Crit	+3		
HotBlooded	+5 - - -	00-		
Frenzy Res	+5 Tenderizer	+3		
Frenzy Res	+4 Pellet Up	-2000		
Biology	+3 Rec Speed	+6000		
Displaying: Rarity				
Dragon Talisman	RARE 7	Value 1400z		

Monster Hunter Generations (Nintendo 3DS)

Tables!

Objectives

- **Include** predictive image correction for **camera images** and elaborate designs

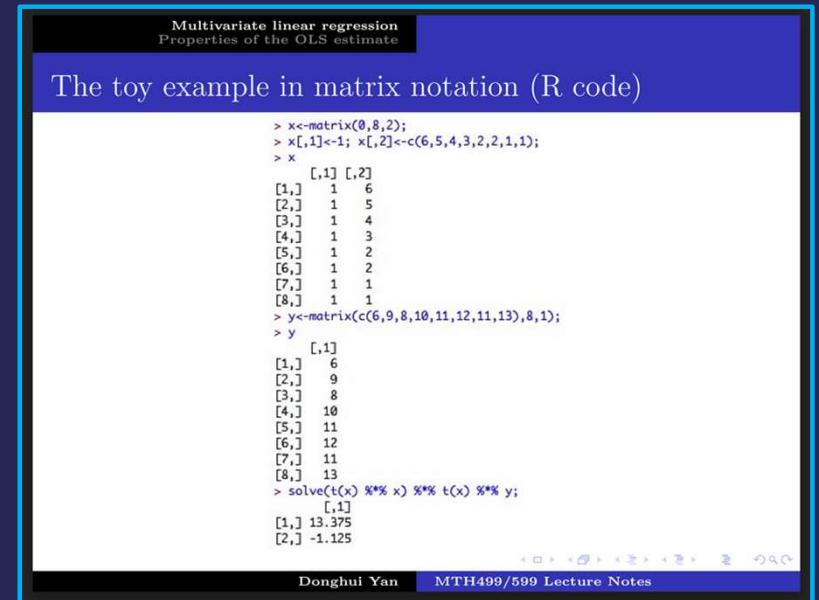


```
Multivariate linear regression
Properties of the OLS estimate

The toy example in matrix notation (R code)

> x<-matrix(0,8,2);
> x[,1]<-1; x[,2]<-c(6,5,4,3,2,2,1,1);
> x
 [,1] [,2]
[1,] 1 6
[2,] 1 5
[3,] 1 4
[4,] 1 3
[5,] 1 2
[6,] 1 2
[7,] 1 1
[8,] 1 1
> y<-matrix(c(6,9,8,10,11,12,11,13),8,1);
> y
 [,1]
[1,] 6
[2,] 9
[3,] 8
[4,] 10
[5,] 11
[6,] 12
[7,] 11
[8,] 13
> solve(x %*% x) %*% t(x) %*% y;
 [,1]
[1,] 13.375
[2,] -1.125

Donghui Yan MTH499/599 Lecture Notes
```



```
Multivariate linear regression
Properties of the OLS estimate

The toy example in matrix notation (R code)

> x<-matrix(0,8,2);
> x[,1]<-1; x[,2]<-c(6,5,4,3,2,2,1,1);
> x
 [,1] [,2]
[1,] 1 6
[2,] 1 5
[3,] 1 4
[4,] 1 3
[5,] 1 2
[6,] 1 2
[7,] 1 1
[8,] 1 1
> y<-matrix(c(6,9,8,10,11,12,11,13),8,1);
> y
 [,1]
[1,] 6
[2,] 9
[3,] 8
[4,] 10
[5,] 11
[6,] 12
[7,] 11
[8,] 13
> solve(t(x) %*% x) %*% t(x) %*% y;
 [,1]
[1,] 13.375
[2,] -1.125

Donghui Yan MTH499/599 Lecture Notes
```

Clean!

Objectives

- Include predictive image correction for camera images and elaborate designs



Baldur's Gate II: Shadows of Amn (PC)

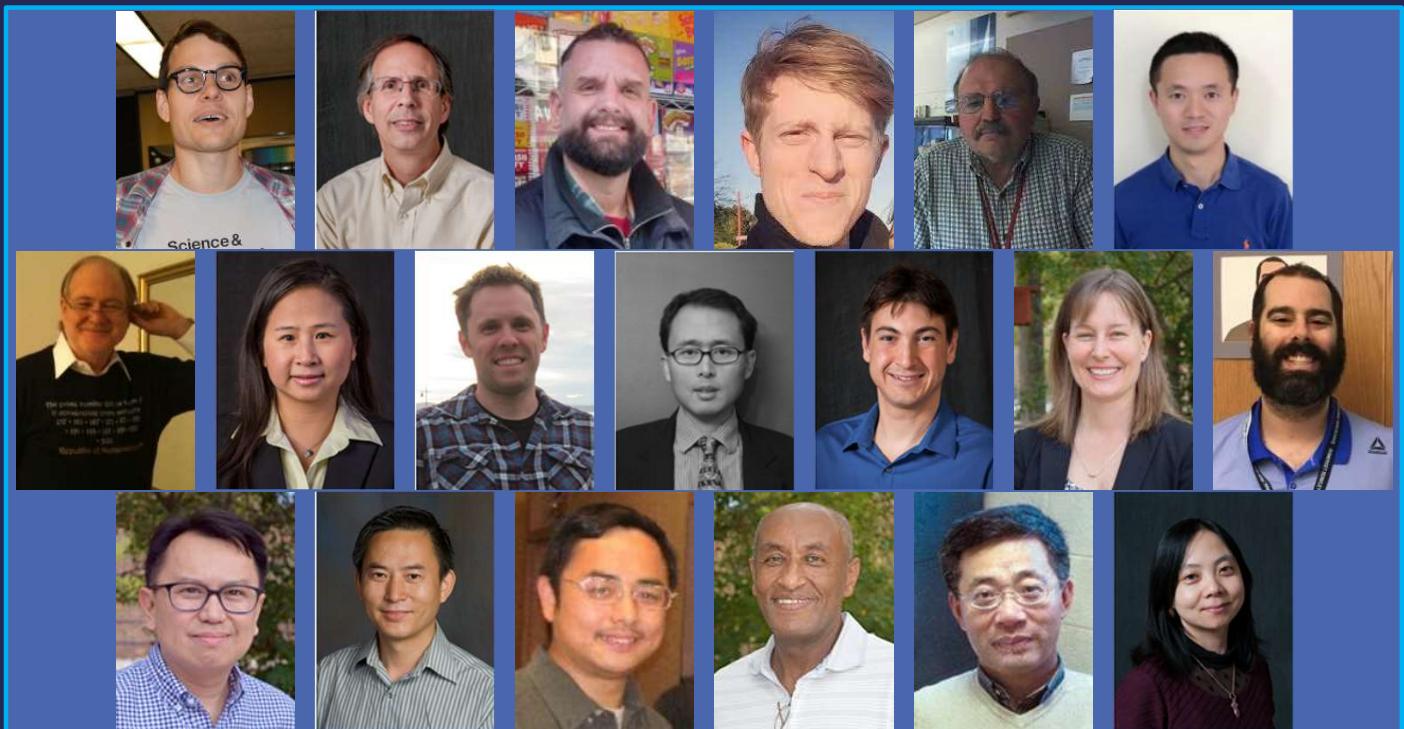


Pokémon: Let's Go, Pikachu! and Let's Go, Eevee! (Nintendo Switch)

Platforms and Awesome People



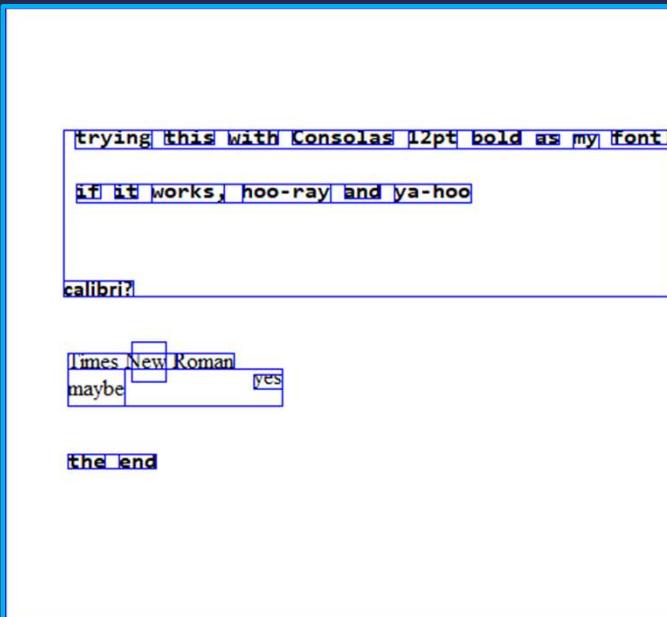
Tesseract OCR



Part 2: Understanding Tesseract OCR

Simple Word Scan

Representation of Word Locations

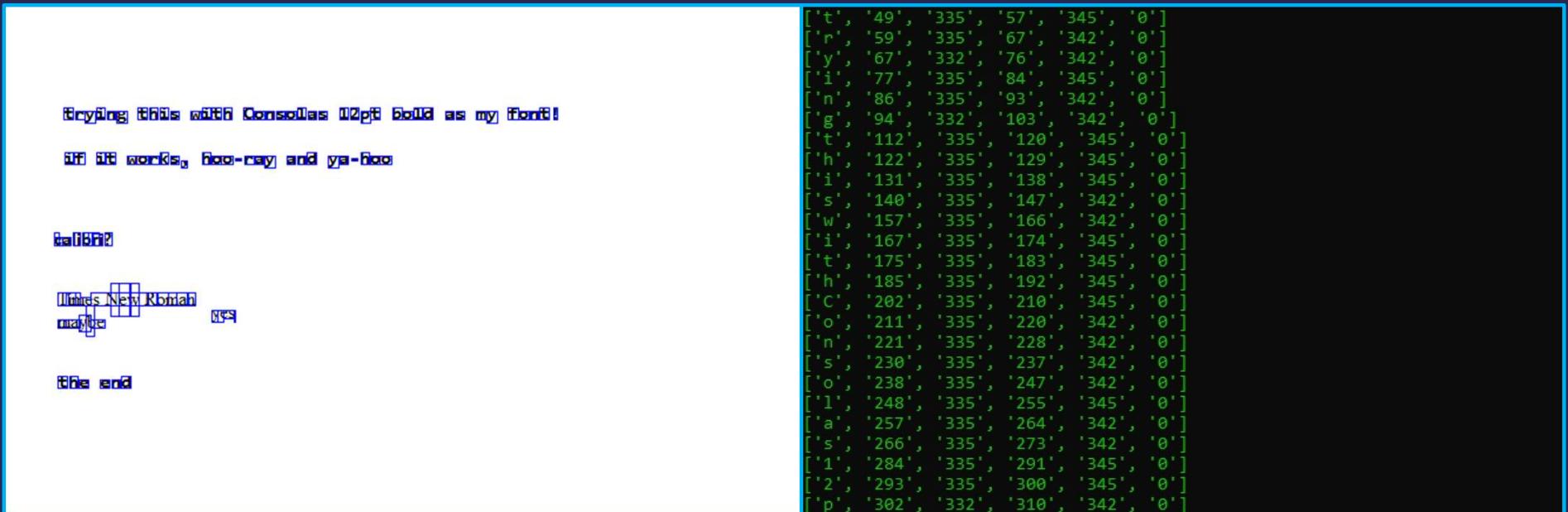


Data from `pytesseract.image_to_data()`

```
[1, '1', '0', '0', '0', '0', '0', '0', '679', '431', '-1', ''],
[2, '1', '1', '0', '0', '0', '41', '86', '428', '117', '-1', ''],
[3, '1', '1', '1', '0', '0', '41', '86', '428', '117', '-1', ''],
[4, '1', '1', '1', '1', '0', '49', '86', '420', '13', '-1', ''],
[5, '1', '1', '1', '1', '1', '49', '86', '54', '13', '93', 'trying'],
[5, '1', '1', '1', '1', '2', '112', '86', '35', '10', '96', 'this'],
[5, '1', '1', '1', '1', '3', '157', '86', '35', '10', '93', 'with'],
[5, '1', '1', '1', '1', '4', '202', '86', '71', '10', '92', 'Consolas'],
[5, '1', '1', '1', '1', '5', '284', '86', '34', '13', '91', '12pt'],
[5, '1', '1', '1', '1', '6', '329', '86', '34', '10', '96', 'bold'],
[5, '1', '1', '1', '1', '7', '374', '89', '16', '7', '96', 'as'],
[5, '1', '1', '1', '1', '8', '400', '89', '18', '10', '96', 'my'],
[5, '1', '1', '1', '1', '9', '427', '86', '42', '10', '96', 'font!'],
[4, '1', '1', '1', '2', '0', '50', '124', '278', '13', '-1', ''],
[5, '1', '1', '1', '2', '1', '50', '124', '17', '10', '93', 'if'],
[5, '1', '1', '1', '2', '2', '77', '124', '16', '10', '94', 'it'],
[5, '1', '1', '1', '2', '3', '103', '124', '51', '13', '93', 'works'],
[5, '1', '1', '1', '2', '4', '167', '124', '62', '13', '92', 'hoo-ray'],
[5, '1', '1', '1', '2', '5', '239', '124', '25', '10', '93', 'and'],
[5, '1', '1', '1', '2', '6', '274', '124', '54', '13', '90', 'ya-hoo'],
[4, '1', '1', '1', '3', '0', '41', '192', '49', '11', '-1', ''],
[5, '1', '1', '1', '3', '1', '41', '192', '49', '11', '57', 'calibri?'],
[2, '1', '2', '0', '0', '0', '44', '243', '117', '11', '-1', ''],
[3, '1', '2', '1', '0', '0', '44', '243', '117', '11', '-1', ''],
[4, '1', '2', '1', '1', '0', '44', '243', '117', '11', '-1', '']]
```

Simple Character Scan

Representation of Character Locations



Block Test 1 – Animal Crossing



Output improves when parsing piece by piece...

Block Test 2 – Monster Hunter

Sharpness	+5 Critical Up	+4
Blunt	+2 Critical Up	+5
Crit Draw	+5 Chain Crit	-1
Sheathing	+10 Snowboron X	+1
Sheathing	+9 Sharpness	-4
Sheathing	+5 Expert	+10
Sheathing	+5 Tenderizer	+4
Sheathe Sharpen	+10 Dreadqueen X	+2
Sheathe Sharpen	+7 Expert	+8

Tg
ert
Sheathing
SE)
Sheathing
Sheathing
Sheathing
Sheath Sharpen
Sheath Sharpen
+2 Critical Up
+5 Chain Crit
Baio cig
SSL tak
Sen

Secrets
BUCS
Baie os
be)
a a
Saal
a
al
Br Ot
cele
Ss) ad
♀

Sheathing +10 Snowboron X +1

Sheathing +10 Srowboron X Saad



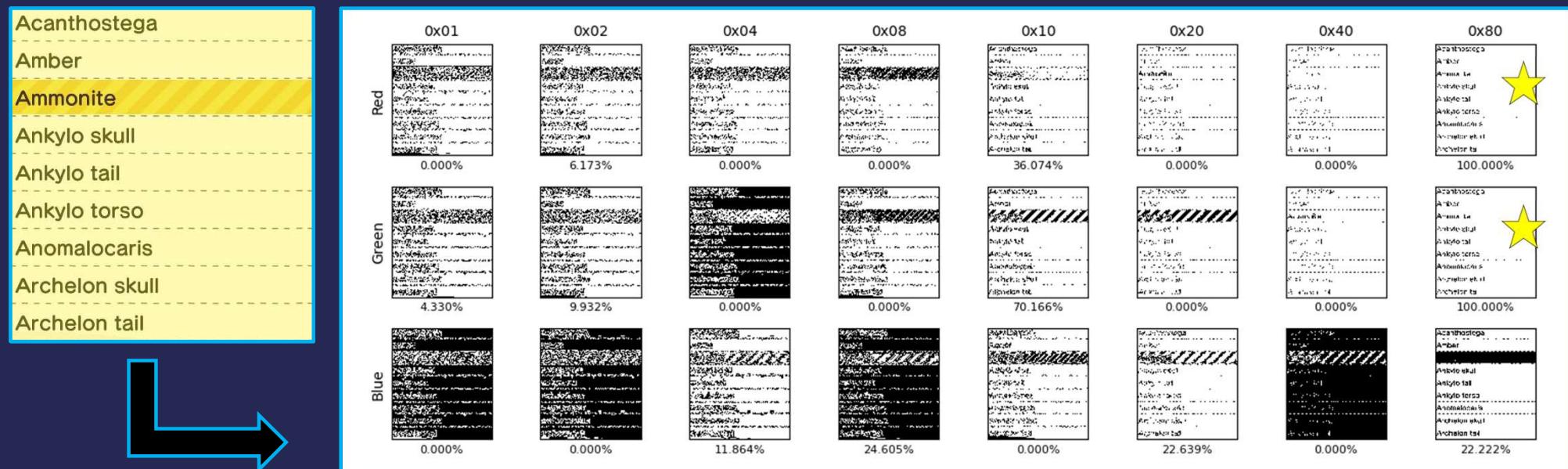
...But sometimes that's still not enough!

Part 3: Image Transformations

(the ones that worked, anyway)

Binarization and Thresholding

- **Binarization:** set pixels to black or white based on brightness (black text on white is best)
- **Thresholding:** assign different brightness cutoffs for binarization



Rotation and Skewing – Matching

- Detect and compare similar feature positions; requires similar details in both images

The toy example in matrix notation (R code)

```
> x<-matrix(0,8,2);
> x[,1]<-1; x[,2]<-c(6,5,4,3,2,2,1,1);
> x
 [,1] [,2]
[1,] 1 6
[2,] 1 5
[3,] 1 4
[4,] 1 3
[5,] 1 2
[6,] 1 2
[7,] 1 1
[8,] 1 1
> y<-matrix(c(6,9,8,10,11,12,11,13),8,1);
> y
 [,1]
[1,] 6
[2,] 9
[3,] 8
[4,] 10
[5,] 11
[6,] 12
[7,] 11
[8,] 13
> solve(t(x) %*% x) %*% t(x) %*% y;
 [,1]
[1,] 13.375
[2,] -1.125
```

Multivariate linear regression
Properties of the OLS estimate

The toy example in matrix notation (R code)

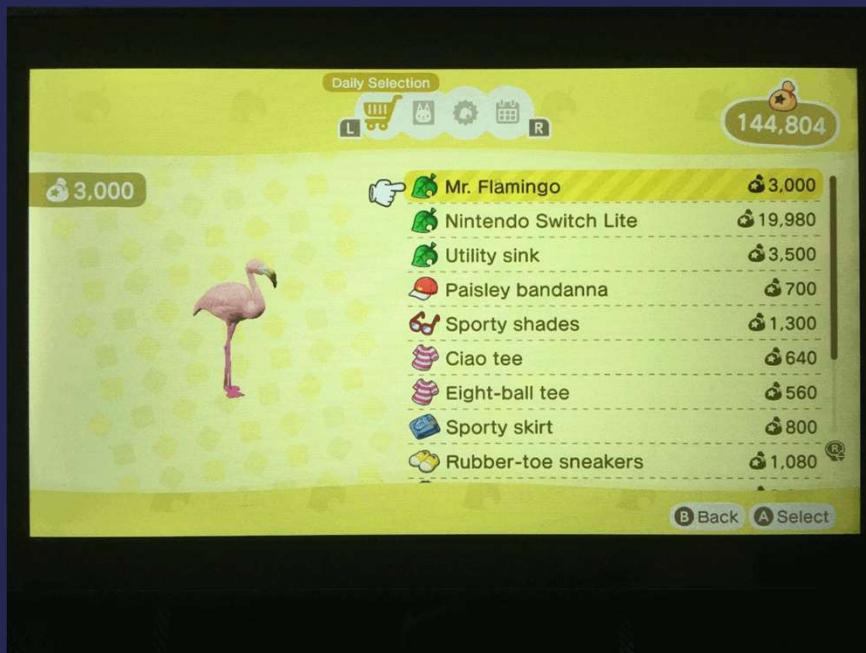
```
> x<-matrix(0,8,2);
> x[,1]<-1; x[,2]<-c(6,5,4,3,2,2,1,1);
> x
 [,1] [,2]
[1,] 1 6
[2,] 1 5
[3,] 1 4
[4,] 1 3
[5,] 1 2
[6,] 1 2
[7,] 1 1
[8,] 1 1
> y<-matrix(c(6,9,8,10,11,12,11,13),8,1);
> y
 [,1]
[1,] 6
[2,] 9
[3,] 8
[4,] 10
[5,] 11
[6,] 12
[7,] 11
[8,] 13
> solve(t(x) %*% x) %*% t(x) %*% y;
 [,1]
[1,] 13.375
[2,] -1.125
```

Multivariate linear regression
Properties of the OLS estimate

Donghui Yan MTH499/599 Lecture Notes

Rotation and Skewing – Edges

- Uses **Canny edge detection**, **Hough lines**, then transform (requires clear edges)

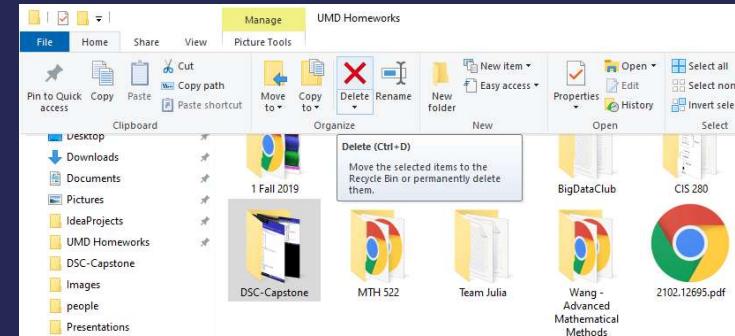


Rotation and Skewing – Hand-holding

- If all else fails, require user input to crop or predetermine row/column spaces



FastCharge	+4	-----	000
FastCharge	+4 Normal Up	+2	000
FastCharge	+3 Bladescale	+1	---
FastCharge	+3 Protection	+8	----
FastCharge	+3 Hero Shield	+2	00-
FastCharge	+3 Light Eater	+4	----
FastCharge	+2 Elderfrost X	+1	----
Stamina	+6 Crit Draw	+2	----
Stamina	+6 Redhelm X	+2	----



Noise Removal

- Eliminate camera noise or background textures – can optionally apply to “empty” cells
- **Moiré patterns** may be introduced from rotation methods

Multivariate linear regression
Properties of the OLS estimate

The toy example in matrix notation (R code)

```
> x<-matrix(c(0,8,2);  
> x[,1]<-1; x[,2]<-c(6,5,4,3,2,2,1,1);  
> x  
[1,] [2]  
[1,] 1 6  
[2,] 1 5  
[3,] 1 4  
[4,] 1 3  
[5,] 1 2  
[6,] 1 2  
[7,] 1 1  
[8,] 1 1  
> y<-matrix(c(6,9,8,10,11,12,11,13),8,1);  
> y  
[1,] 6  
[2,] 9  
[3,] 8  
[4,] 10  
[5,] 11  
[6,] 12  
[7,] 11  
[8,] 13  
> solve(t(x) %*% x) %*% t(x) %*% y;  
[1,]  
[1,] 13.375  
[2,] -1.125
```

Donghui Yan MTH499/599 Lecture Notes



Multivariate linear regression
Properties of the OLS estimate

The toy example in matrix notation (R code)

```
> x<-matrix(c(0,8,2);  
> x[,1]<-1; x[,2]<-c(6,5,4,3,2,2,1,1);  
> x  
[1,] [2]  
[1,] 1 6  
[2,] 1 5  
[3,] 1 4  
[4,] 1 3  
[5,] 1 2  
[6,] 1 2  
[7,] 1 1  
[8,] 1 1  
> y<-matrix(c(6,9,8,10,11,12,11,13),8,1);  
> y  
[1,] 6  
[2,] 9  
[3,] 8  
[4,] 10  
[5,] 11  
[6,] 12  
[7,] 11  
[8,] 13  
> solve(t(x) %*% x) %*% t(x) %*% y;  
[1,]  
[1,] 13.375  
[2,] -1.125
```

Donghui Yan MTH499/599 Lecture Notes



Part 4: Dictionary-based Text Correction

Dictionary Implementation

- Use pre-defined dictionary for given location, then use **Levenshtein distance** to match
- Extremely effective but requires additional user input

Sharpness	+5	Critical Up	+4	OO-
Blunt	+2	Critical Up	+5	OO-
Crit Draw	+5	Chain Crit	-1	OOO
Sheathing	+10	Snowbaron X	+1	OO-
Sheathing	+9	Sharpness	-4	OOO
Sheathing	+5	Expert	+10	O---
Sheathing	+5	Tenderizer	+4	O--
Sheathe Sharpen	+10	Dreadqueen X	+2	OO-
Sheathe Sharpen	+7	Expert	+8	OO-



Sharpness	45	Ceticol Up	+4	oo
Blunt	+2	Crticolo Up	+5	oo
Crk Drow	45	Choin Grit	1	oo
Sheathing	+10	Snowbaron X	+1	oo
Steathing	9	Shorenness	-4	oo
Steathing	+5	Expert	+10	oo
Steahing	+5	Tenderizer	+4	oo
Sheathe Sherpen	+10	Drecsueen X	+2	oo
Sheathe Sharpen	+7	Expert	+8	oo

No dictionary:
86.41% accuracy

89.74% accuracy
for word columns

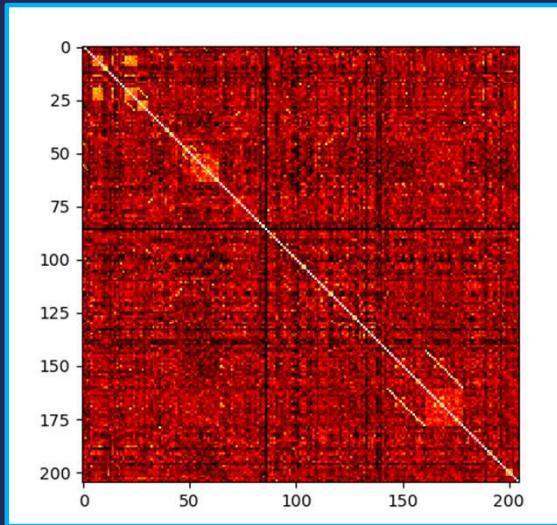
Sharpness	-5	Critical Up	+4	oo-
Blunt	+2	Critical Up	+5	oo-
Crit Draw	-5	Chain Crit	-1	oo-
Sheathing	+10	Snowbaron X	+1	oo-
Sheathing	-9	Sharpness	-4	oo-
Sheathing	+5	Expert	+10	oo-
Sheathing	+5	Tenderizer	+4	oo-
Sheathe Sharpen	+10	Dreadqueen X	+2	oo-
Sheathe Sharpen	+7	Expert	+8	oo-

With dictionary:
92.96% accuracy

100.00% accuracy
for word columns

Dictionary Limits

- Similar words may be misinterpreted ("couch" vs. "conch")
- **Levenshtein matrix** estimates error likelihood



MHGU Skill Names:
205 unique items
11.10% mean distance

Thunderlord	Thunderlord X	0.916667
Wind Res	Bind Res	0.875000
Fencing	D. Fencing	0.823529
Poison	Poison C+	0.800000
Hearing	Hoardings	0.800000
Dead Eye	Deadeye	0.800000
Loading	Hoardings	0.800000
Pierce Up	Pierce S+	0.777778
Frenzy Res	Fire Res	0.777778
Sharpener	Sharpness	0.777778



Very common for one term to extend another!

Dictionary Limits



Mr. Flamingo

stone tablet

peach chair

music stand

ironwood chair

diner mini table

wooden-block bed

wooden-block bed

ironwood low table

wooden-block stool

Mrs. Flamingo

stone table

beach chair

mic stand

ironwood cart

diner dining table

wooden-block table

wooden-block bench

ironwood table

wooden-block stereo

0.960000

0.956522

0.909091

0.900000

0.888889

0.882353

0.882353

0.882353

0.875000

0.864865

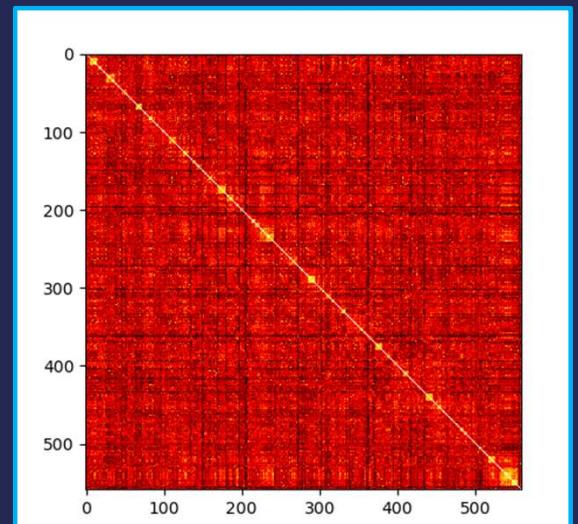
Could be a
problem!



ACNH Houseware Names:

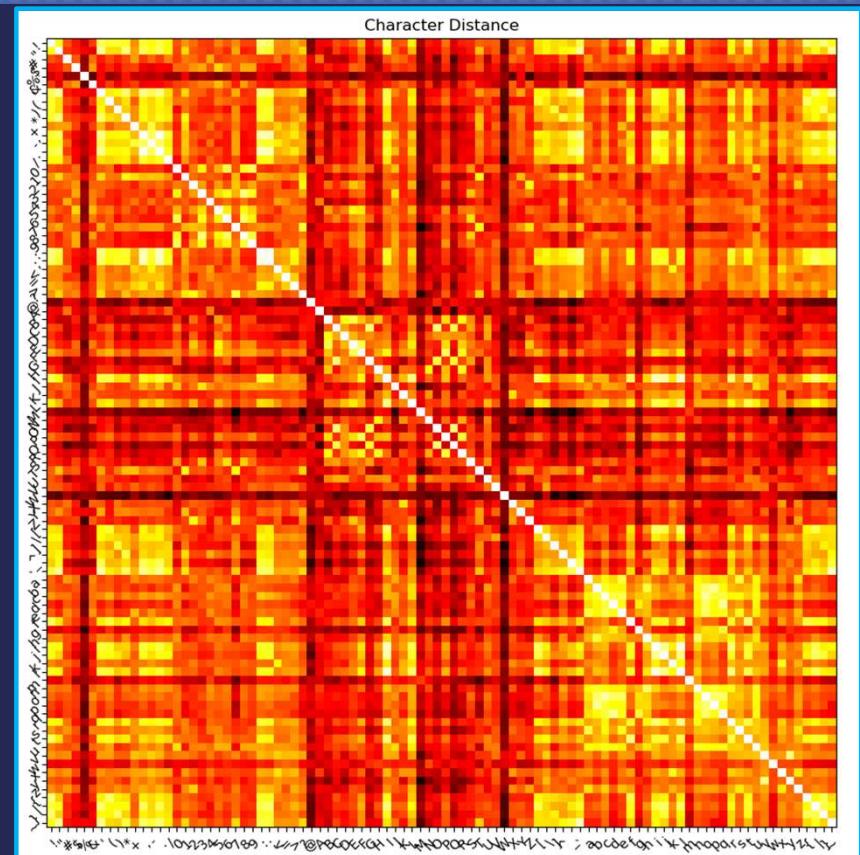
559 unique items
29.09% mean distance

More items to compare
means greater error risk!



Font Character-Shape Tiebreaker

- When Levenshtein fails, compare the shapes of the characters themselves!
 - Import font or compare with a common one



Part 5: Conclusion, why not

Yes, it works!

Issues and considerations:

- Currently requires hand-holding
 - Not fully autonomous AI
- More testing on different images
- Method refinement
- Convolutional Neural Networks

Could add:

- “Spot the differences” features
- Unstructured list collection
- Software creation and UI



Questions?

(that's it)