# Analysis of Protest Imagery Workshop

Michelle Torres
smtorres@ucla.edu
UCLA

October 1, 2024

Welcome!

Welcome!

Thank you for being here

# Goals

1. Cover the basics of images-as-data

# GOALS

1. Cover the basics of images-as-data
2. Introduce you to canonical and recent(-ish) computational methods and tools for the analysis of imagery

## GOALS

1. Cover the basics of images-as-data
2. Introduce you to canonical and recent(-ish) computational methods and tools for the analysis of imagery
3. FOCUS ON:

# GOALS

1. Cover the basics of images-as-data
2. Introduce you to canonical and recent(-ish) computational methods and tools for the analysis of imagery
3. FOCUS ON:
   - Logic, statistical foundations, and structure

# GOALS

1. Cover the basics of images-as-data
2. Introduce you to canonical and recent(-ish) computational methods and tools for the analysis of imagery
3. FOCUS ON:
    - Logic, statistical foundations, and structure
    - Implementation

# GOALS

1. Cover the basics of images-as-data
2. Introduce you to canonical and recent(-ish) computational methods and tools for the analysis of imagery
3. FOCUS ON:
    - Logic, statistical foundations, and structure
    - Implementation
4. Practice some coding in `Python` and `R`

# Goals

1. Cover the basics of images-as-data
2. Introduce you to canonical and recent(-ish) computational methods and tools for the analysis of imagery
3. FOCUS ON:
    - Logic, statistical foundations, and structure
    - Implementation
4. Practice some coding in `Python` and `R`
5. Learn CS tools through our SS glasses

# GOALS

1. Cover the basics of images-as-data
2. Introduce you to canonical and recent(-ish) computational methods and tools for the analysis of imagery
3. FOCUS ON:
   - Logic, statistical foundations, and structure
   - Implementation
4. Practice some coding in `Python` and `R`
5. Learn CS tools through our SS glasses
   - Why cares (or should care) about images?

# Goals

1. Cover the basics of images-as-data
2. Introduce you to canonical and recent(-ish) computational methods and tools for the analysis of imagery
3. FOCUS ON:
   - Logic, statistical foundations, and structure
   - Implementation
4. Practice some coding in `Python` and `R`
5. Learn CS tools through our SS glasses
   - Why cares (or should care) about images?
   - Responsible and rigorous use of “flashy” and “glittery” tools

# Goals

1. Cover the basics of images-as-data
2. Introduce you to canonical and recent(-ish) computational methods and tools for the analysis of imagery
3. FOCUS ON:
   - Logic, statistical foundations, and structure
   - Implementation
4. Practice some coding in `Python` and `R`
5. Learn CS tools through our SS glasses
   - Why cares (or should care) about images?
   - Responsible and rigorous use of “flashy” and “glittery” tools
6. Have fun! (Yes, yes, I know I am biased!)

Let’s start!

# Why images?

- Images are powerful: extra information + emotional activation + *see to believe* = recall, engagement, attitude formation

# Why images?

- Images are powerful: extra information + emotional activation + *see to believe* = recall, engagement, attitude formation
- Images are (kind of) universal (e.g. compare them to spoken languages)

## Why images?

- Images are powerful: extra information + emotional activation + *see to believe* = recall, engagement, attitude formation
- Images are (kind of) universal (e.g. compare them to spoken languages)
- Visuals are frames

# Why images of protests?

# Why images of protests?

- We have “missing protest data”: bias in coverage, hard-to-measure elements → Social media as a new source of data (Pan & Zhang 2020, Steinert-Threlkeld 2023)

# Why images of protests?

- We have "missing protest data": bias in coverage, hard-to-measure elements → Social media as a new source of data (Pan & Zhang 2020, Steinert-Threlkeld 2023)
- Study coverage bias/framing and factors behind it (Torres 2023)


# WHY IMAGES OF PROTESTS?

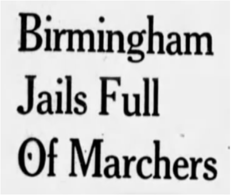
- We have “missing protest data”: bias in coverage, hard-to-measure elements → Social media as a new source of data (Pan & Zhang 2020, Steinert-Threlkeld 2023)
- Study coverage bias/framing and factors behind it (Torres 2023)
- Measure difficult traits: magnitude, protest infrastructure and dynamics (Steinert-Threlkeld, et al., Fruhstorfer and Kittel)

# Why images of protests?

- We have “missing protest data”: bias in coverage, hard-to-measure elements → Social media as a new source of data (Pan & Zhang 2020, Steinert-Threlkeld 2023)
- Study coverage bias/framing and factors behind it (Torres 2023)
- Measure difficult traits: magnitude, protest infrastructure and dynamics (Steinert-Threlkeld, et al., Fruhstorfer and Kittel)
- Explore other important characteristics: emotions, violence, strategies (Casas & Webb Williams 2019)

# WHY IMAGES OF PROTESTS?

- We have “missing protest data”: bias in coverage, hard-to-measure elements → Social media as a new source of data (Pan & Zhang 2020, Steinert-Threlkeld 2023)
- Study coverage bias/framing and factors behind it (Torres 2023)
- Measure difficult traits: magnitude, protest infrastructure and dynamics (Steinert-Threlkeld, et al., Fruhstorfer and Kittel)
- Explore other important characteristics: emotions, violence, strategies (Casas & Webb Williams 2019)
- Impactful way of communicating a message


Birmingham Jails Full Of Marchers

## Why computers?

- Process large pools of images

# WHY COMPUTERS?

- Process large pools of images
- Increase consistency/reliability and decrease bias (*)

# WHY COMPUTERS?

- Process large pools of images
- Increase consistency/reliability and decrease bias (*)
- Helping humans to “see” and discover (*)

# Why computers?

- Process large pools of images
- Increase consistency/reliability and decrease bias (*)
- Helping humans to “see” and discover (*)
- Computer vision: Teaching computers to see

# IMAGE AS DATA: TEACHING THE COMPUTER TO SEE

# Image as Data: Teaching the computer to see

**A very hard task!**

## IMAGE AS DATA: TEACHING THE COMPUTER TO SEE

### A very hard task!

- Computers are great at following instructions reliably...

## A very hard task!

- Computers are great at following instructions reliably...
- ... but they are bad at inferences

## A very hard task!

- Computers are great at following instructions reliably...
- ... but they are bad at inferences

# IMAGE AS DATA: TEACHING THE COMPUTER TO SEE

## A very hard task!

- Computers are great at following instructions reliably...
- ... but they are bad at inferences

## A very hard task!

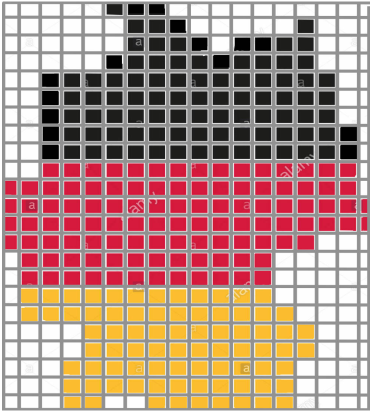
- Computers are great at following instructions reliably...
- ... but they are bad at inferences

## Getting ready

- Course website, Github: smtorres/FU_Workshop
- Google Colab notebooks
  - Notebook 1: here
  - Notebook 2: here
  - Notebook 3: here
- Follow instructions here
- When doing your own projects:
  - Install Keras (here), with `tensorflow` backend
  - Install the following python libraries: numpy, scipy, cv2, matplotlib, PIL, sklearn ⇒ Look for tutorials for your machine
    - Check tutorials for OpenCV installation here
    - I suggest OpenCV 3.X and its compilation from source for full functionality

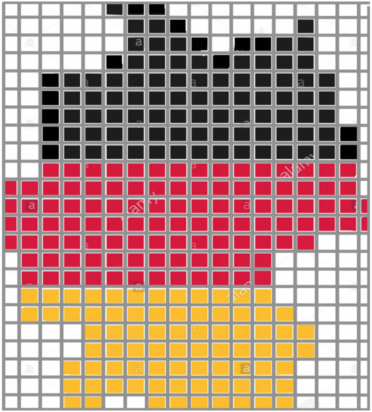
# Image basics

- An image is a set of pixels:



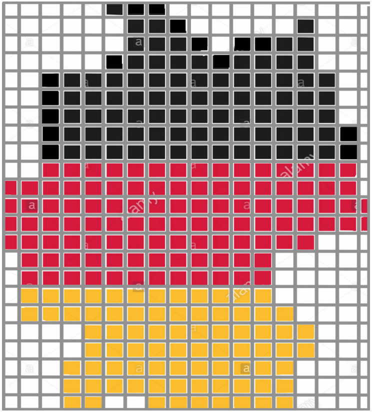
# IMAGE BASICS

- An image is a set of pixels:
  - Finest unit (defines height and width)



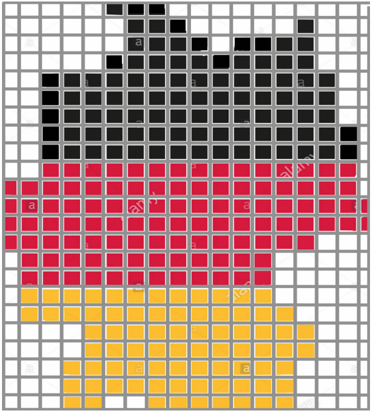
# IMAGE BASICS

- An image is a set of pixels:
  - Finest unit (defines height and width)
  - Grayscale: intensity of light, Color: color intensity per channel.



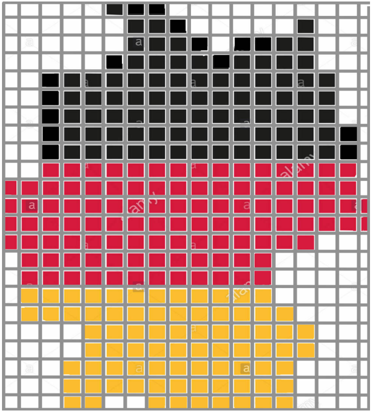
# IMAGE BASICS

- An image is a set of pixels:
  - Finest unit (defines height and width)
  - Grayscale: intensity of light, Color: color intensity per channel.
- Matrix representation



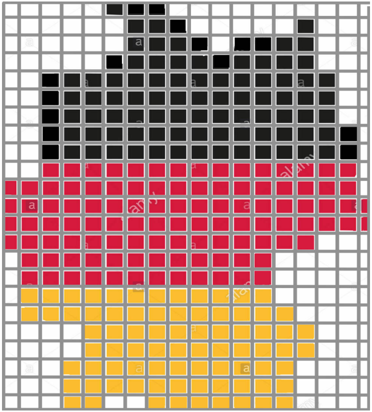
# IMAGE BASICS

- An image is a set of pixels:
  - Finest unit (defines height and width)
  - Grayscale: intensity of light, Color: color intensity per channel.
- Matrix representation
  - Grayscale: one matrix



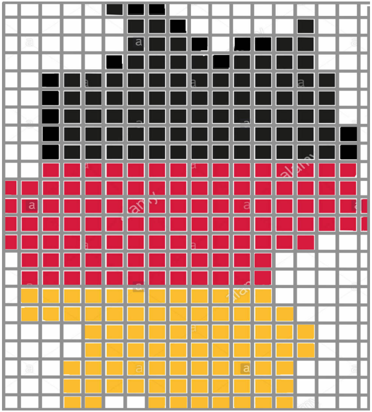
# IMAGE BASICS

- An image is a set of pixels:
  - Finest unit (defines height and width)
  - Grayscale: intensity of light, Color: color intensity per channel.
- Matrix representation
  - Grayscale: one matrix
  - Color: array with a matrix for each color channel (Red, Green, and Blue)



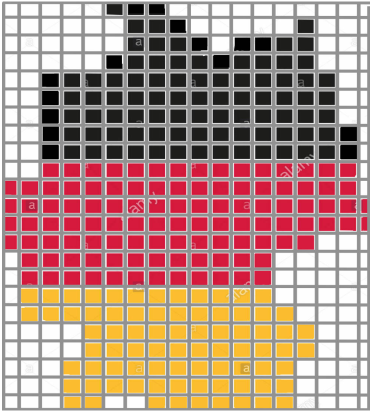
# IMAGE BASICS

- An image is a set of pixels:
  - Finest unit (defines height and width)
  - Grayscale: intensity of light, Color: color intensity per channel.
- Matrix representation
  - Grayscale: one matrix
  - Color: array with a matrix for each color channel (Red, Green, and Blue)
- Notice that in OpenCV:



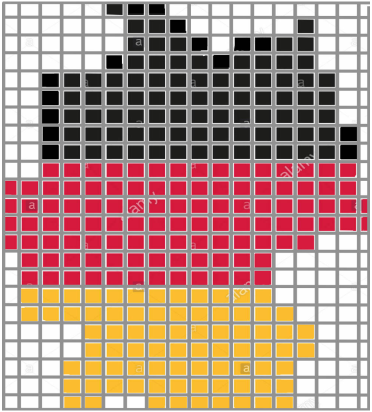
# IMAGE BASICS

- An image is a set of pixels:
  - Finest unit (defines height and width)
  - Grayscale: intensity of light, Color: color intensity per channel.
- Matrix representation
  - Grayscale: one matrix
  - Color: array with a matrix for each color channel (Red, Green, and Blue)
- Notice that in OpenCV:
  - Color channel specification is BRG instead of RGB



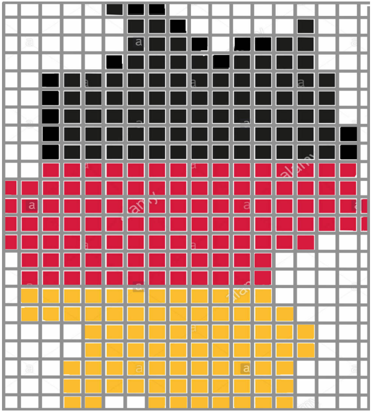
# IMAGE BASICS

- An image is a set of pixels:
  - Finest unit (defines height and width)
  - Grayscale: intensity of light, Color: color intensity per channel.
- Matrix representation
  - Grayscale: one matrix
  - Color: array with a matrix for each color channel (Red, Green, and Blue)
- Notice that in OpenCV:
  - Color channel specification is BRG instead of RGB
  - Origin of image is different (top left corner)



# Image basics

- An image is a set of pixels:
  - Finest unit (defines height and width)
  - Grayscale: intensity of light, Color: color intensity per channel.
- Matrix representation
  - Grayscale: one matrix
  - Color: array with a matrix for each color channel (Red, Green, and Blue)
- Notice that in OpenCV:
  - Color channel specification is BRG instead of RGB
  - Origin of image is different (top left corner)
  - In `numpy` you specify the $y-$coordinates of an image first: `X2 = image[y0:y1, x0:x1]`



## Describing an image

- Think about the “tokens” or elements that give meaning to text. What are those?

## DESCRIBING AN IMAGE

- Think about the “tokens” or elements that give meaning to text. What are those?
- The main challenge with images: a lot of pixels that mostly make sense when analyzed in clusters and not as units.

# DESCRIBING AN IMAGE

- Think about the “tokens” or elements that give meaning to text. What are those?
- The main challenge with images: a lot of pixels that mostly make sense when analyzed in clusters and not as units.
- Therefore, we use **image descriptors** to characterize the content on an image ***globally*** or feature descriptors to locally quantify ***regions*** of the image.

# DESCRIBING AN IMAGE

- Think about the "tokens" or elements that give meaning to text. What are those?
- The main challenge with images: a lot of pixels that mostly make sense when analyzed in clusters and not as units.
- Therefore, we use **image descriptors** to characterize the content on an image ***globally*** or feature descriptors to locally quantify ***regions*** of the image.
  - Color

## Describing an image

- Think about the “tokens” or elements that give meaning to text. What are those?
- The main challenge with images: a lot of pixels that mostly make sense when analyzed in clusters and not as units.
- Therefore, we use **image descriptors** to characterize the content on an image ***globally*** or feature descriptors to locally quantify ***regions*** of the image.
  - Color
  - Texture

# Describing an image

- Think about the "tokens" or elements that give meaning to text. What are those?
- The main challenge with images: a lot of pixels that mostly make sense when analyzed in clusters and not as units.
- Therefore, we use **image descriptors** to characterize the content on an image ***globally*** or feature descriptors to locally quantify ***regions*** of the image.
  - Color
  - Texture
  - Shape

# Describing an image

- Think about the “tokens” or elements that give meaning to text. What are those?
- The main challenge with images: a lot of pixels that mostly make sense when analyzed in clusters and not as units.
- Therefore, we use **image descriptors** to characterize the content on an image ***globally*** or feature descriptors to locally quantify ***regions*** of the image.
  - Color
  - Texture
  - Shape
  - Pixel intensity change

# Describing an image

- Think about the "tokens" or elements that give meaning to text. What are those?
- The main challenge with images: a lot of pixels that mostly make sense when analyzed in clusters and not as units.
- Therefore, we use **image descriptors** to characterize the content on an image ***globally*** or feature descriptors to locally quantify ***regions*** of the image.
  - Color
  - Texture
  - Shape
  - Pixel intensity change
  - Edges, objects, etc.

# Describing an image

- Think about the “tokens” or elements that give meaning to text. What are those?
- The main challenge with images: a lot of pixels that mostly make sense when analyzed in clusters and not as units.
- Therefore, we use **image descriptors** to characterize the content on an image ***globally*** or feature descriptors to locally quantify ***regions*** of the image.
  - Color
  - Texture
  - Shape
  - Pixel intensity change
  - Edges, objects, etc.
- Feature vectors: A series of numbers used to numerically quantify the contents of an image (or regions of it) ⇒ WE USE THEM TO CREATE TOKENS!

# An example: Color statistics

### Channel statistics

- Very intuitive and simple
- Basic statistics of each color channel
  1. Separate channels
  2. Compute moments for each channel

### Histograms

- More information based on distribution
- 3D histogram of colors
  1. Convert image to L*a*b color space
  2. Compute 3D histogram

4. Concatenate to form *feature vector*

Voilá! You have a global descriptor for your image

# An example: Color statistics

**Channel statistics**

- Very intuitive and simple
- Basic statistics of each color channel
  1. Separate channels
  2. Compute moments for each channel

**Histograms**

- More information based on distribution
- 3D histogram of colors
  1. Convert image to L*a*b color space
  2. Compute 3D histogram

4. Concatenate to form *feature vector*

Voilá! You have a global descriptor for your image → Your feature vector = a token!

# COLOR STATISTICS, CONT.

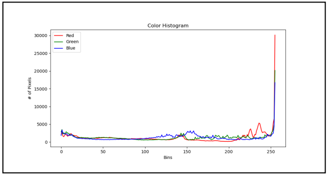
# COLOR STATISTICS, CONT.

# COLOR STATISTICS, CONT.


| Channel | Mean | Median | Std. Dev |
|---|---|---|---|
| Red | 135.4 | 136.5 | 77.2 |
| Green | 143 | 146 | 85.7 |
| Blue | 147 | 146 | 90.8 |

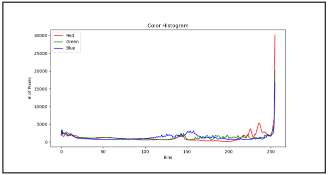
# COLOR STATISTICS, CONT.

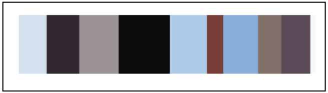

| Channel | Mean | Median | Std. Dev |
|---|---|---|---|
| Red | 135.4 | 136.5 | 77.2 |
| Green | 143 | 146 | 85.7 |
| Blue | 147 | 146 | 90.8 |



# Color statistics, cont.


| Channel | Mean | Median | Std. Dev |
|---|---|---|---|
| Red | 135.4 | 136.5 | 77.2 |
| Green | 143 | 146 | 85.7 |
| Blue | 147 | 146 | 90.8 |



## And then what do we to with those tokens?

- Stop thinking about images for a second

## And then what do we to with those tokens?

- Stop thinking about images for a second
- What do you guys study or research?

# And then what do we to with those tokens?

- Stop thinking about images for a second
- What do you guys study or research?
- You may have:

## And then what do we to with those tokens?

- Stop thinking about images for a second
- What do you guys study or research?
- You may have:
    - Dataframes with variables in columns and observations in rows

## And then what do we to with those tokens?

- Stop thinking about images for a second
- What do you guys study or research?
- You may have:
  - Dataframes with variables in columns and observations in rows
  - Interviews with text

## AND THEN WHAT DO WE TO WITH THOSE TOKENS?

- Stop thinking about images for a second
- What do you guys study or research?
- You may have:
  - Dataframes with variables in columns and observations in rows
  - Interviews with text
  - Paragraphs of speeches from MPs

## And then what do we to with those tokens?

- Stop thinking about images for a second
- What do you guys study or research?
- You may have:
  - Dataframes with variables in columns and observations in rows
  - Interviews with text
  - Paragraphs of speeches from MPs
  - ...

# And then what do we to with those tokens?

- Stop thinking about images for a second
- What do you guys study or research?
- You may have:
  - Dataframes with variables in columns and observations in rows
  - Interviews with text
  - Paragraphs of speeches from MPs
  - ...
- And what do you do with that?

## And then what do we to with those tokens?

- Stop thinking about images for a second
- What do you guys study or research?
- You may have:
  - Dataframes with variables in columns and observations in rows
  - Interviews with text
  - Paragraphs of speeches from MPs
  - ...
- And what do you do with that?
  - Run regressions to make inferences

# And then what do we to with those tokens?

- Stop thinking about images for a second
- What do you guys study or research?
- You may have:
  - Dataframes with variables in columns and observations in rows
  - Interviews with text
  - Paragraphs of speeches from MPs
  - ...
- And what do you do with that?
  - Run regressions to make inferences
  - Predict values of interest

# And then what do we to with those tokens?

- Stop thinking about images for a second
- What do you guys study or research?
- You may have:
  - Dataframes with variables in columns and observations in rows
  - Interviews with text
  - Paragraphs of speeches from MPs
  - ...
- And what do you do with that?
  - Run regressions to make inferences
  - Predict values of interest
  - Identify topics in a corpus of texts

## AND THEN WHAT DO WE TO WITH THOSE TOKENS?

- Stop thinking about images for a second
- What do you guys study or research?
- You may have:
  - Dataframes with variables in columns and observations in rows
  - Interviews with text
  - Paragraphs of speeches from MPs
  - ...
- And what do you do with that?
  - Run regressions to make inferences
  - Predict values of interest
  - Identify topics in a corpus of texts
  - ...

# EXAMPLE: SIMILARITY/DISTANCE TO IMAGE BENCHMARK

## Example: similarity/distance to image benchmark

- Motivation: colors in an image might trigger different emotional reactions (Pro-tip: read about color theory!)

## Example: similarity/distance to image benchmark

- Motivation: colors in an image might trigger different emotional reactions (Pro-tip: read about color theory!)
- Question: how similar/different are visual stimuli from a benchmark photo?

# EXAMPLE: SIMILARITY/DISTANCE TO IMAGE BENCHMARK

- Motivation: colors in an image might trigger different emotional reactions (Pro-tip: read about color theory!)
- Question: how similar/different are visual stimuli from a benchmark photo?

$$d(p,q) = \sqrt{\sum_{i=1}^{N}(q_i - p_i)^2}$$

# Example: similarity/distance to image benchmark

- Motivation: colors in an image might trigger different emotional reactions (Pro-tip: read about color theory!)
- Question: how similar/different are visual stimuli from a benchmark photo?

$$d(p,q) = \sqrt{\sum_{i=1}^{N}(q_i - p_i)^2}$$

# EXAMPLE: SIMILARITY/DISTANCE TO IMAGE BENCHMARK, CONT.

## A very valid question:

But... aren't these tokens too *simple*?

## A VERY VALID QUESTION:

But... aren't these tokens too *simple*?

- Yes! They are... (still useful depending on your application)

## A VERY VALID QUESTION:

But... aren't these tokens too *simple*?

- Yes! They are... (still useful depending on your application)
- The blessing and curse of images is their complexity and richness

## A VERY VALID QUESTION:

But... aren't these tokens too *simple*?

- Yes! They are... (still useful depending on your application)
- The blessing and curse of images is their complexity and richness
- 1) Existence of and 2) interaction between A LOT of features (remember: texture, shapes, objects, colors, etc.)

## A VERY VALID QUESTION:

But... aren't these tokens too *simple*?

- Yes! They are... (still useful depending on your application)
- The blessing and curse of images is their complexity and richness
- 1) Existence of and 2) interaction between A LOT of features (remember: texture, shapes, objects, colors, etc.)
- So how can we take all those things into account to correctly capture the **content** and (potentially) **message** of our images?

## A VERY VALID QUESTION:

But... aren't these tokens too *simple*?

- Yes! They are... (still useful depending on your application)
- The blessing and curse of images is their complexity and richness
- 1) Existence of and 2) interaction between A LOT of features (remember: texture, shapes, objects, colors, etc.)
- So how can we take all those things into account to correctly capture the **content** and (potentially) **message** of our images?
- In other more ambitious words:

## A VERY VALID QUESTION:

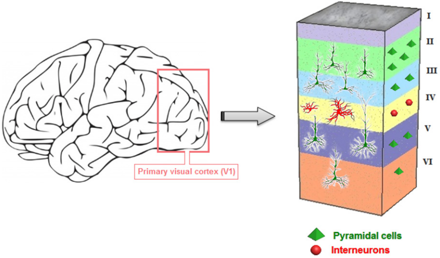
But... aren't these tokens too *simple*?

- Yes! They are... (still useful depending on your application)
- The blessing and curse of images is their complexity and richness
- 1) Existence of and 2) interaction between A LOT of features (remember: texture, shapes, objects, colors, etc.)
- So how can we take all those things into account to correctly capture the **content** and (potentially) **message** of our images?
- In other more ambitious words:

  How do we teach the computer to *see* like us?

# Convolutional Neural Networks

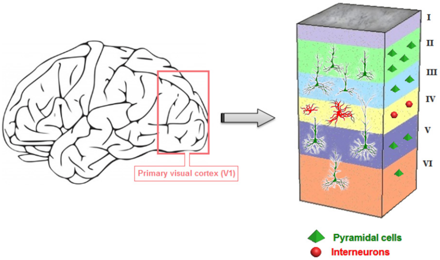
# SEEING LIKE A HUMAN




Credit: *Bachatene, Bharmauria and Molotchnikoff (2012).*

- Light enters through our eyes and internally recreates the imagery that the light forms

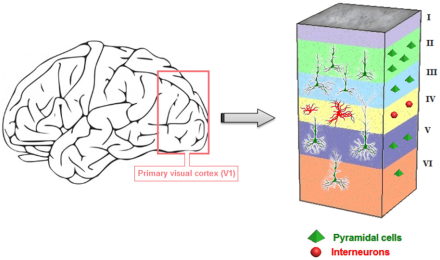
# SEEING LIKE A HUMAN




Credit: *Bachatene, Bharmauria and Molotchnikoff (2012).*

- Light enters through our eyes and internally recreates the imagery that the light forms
- This signal is sent to the brain for analysis

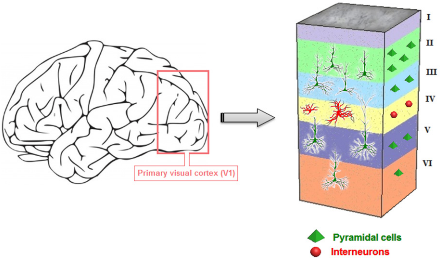
# SEEING LIKE A HUMAN




Credit: *Bachatene, Bharmauria and Molotchnikoff (2012).*

- Light enters through our eyes and internally recreates the imagery that the light forms
- This signal is sent to the brain for analysis
- Neurons are organized into layers.

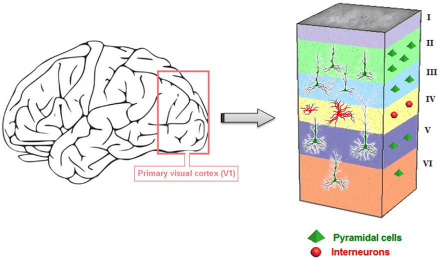
# SEEING LIKE A HUMAN




Credit: *Bachatene, Bharmauria and Molotchnikoff (2012).*

- Light enters through our eyes and internally recreates the imagery that the light forms
- This signal is sent to the brain for analysis
- Neurons are organized into layers.
- Every layer breaks down the signal into small pieces, allowing each of its neurons to focus on a unique piece of information.

# SEEING LIKE A HUMAN




Credit: *Bachatene, Bharmauria and Molotchnikoff (2012).*

- Light enters through our eyes and internally recreates the imagery that the light forms
- This signal is sent to the brain for analysis
- Neurons are organized into layers.
- Every layer breaks down the signal into small pieces, allowing each of its neurons to focus on a unique piece of information.
- The first layers identify basic visual patterns, intermediate layers transform patterns into shapes, and the last layers convert shapes into objects.

## SEEING LIKE A HUMAN, CONT.

- The information from all of these layers allows the brain to *make sense* of what we are seeing

## Seeing like a human, cont.

- The information from all of these layers allows the brain to *make sense* of what we are seeing
- There is an extra crucial step: connecting the processed visual stimulus to a concept or meaning

## Seeing like a human, cont.

- The information from all of these layers allows the brain to *make sense* of what we are seeing
- There is an extra crucial step: connecting the processed visual stimulus to a concept or meaning
- The blurry line of my personal and professional lives: How babies learn to see and identify objects

## Seeing like a human, cont.

- The information from all of these layers allows the brain to *make sense* of what we are seeing
- There is an extra crucial step: connecting the processed visual stimulus to a concept or meaning
- The blurry line of my personal and professional lives: How babies learn to see and identify objects
- Distinguishing, tracking, and naming: “Oh look, a **puppy** is coming!”, “This is a puppy”, “The puppy wants to kiss you”

# Seeing like a human, cont.

- The information from all of these layers allows the brain to *make sense* of what we are seeing
- There is an extra crucial step: connecting the processed visual stimulus to a concept or meaning
- The blurry line of my personal and professional lives: How babies learn to see and identify objects
- Distinguishing, tracking, and naming: “Oh look, a **puppy** is coming!”, “This is a puppy”, “The puppy wants to kiss you”
    - Baby: “Hmhm... this furry, medium, four-legged, moving blob seems to be called **puppy**”

## Seeing like a human, cont.

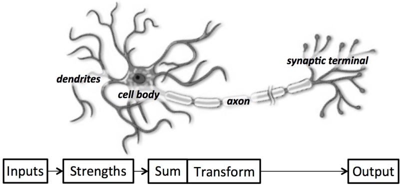
- The information from all of these layers allows the brain to *make sense* of what we are seeing
- There is an extra crucial step: connecting the processed visual stimulus to a concept or meaning
- The blurry line of my personal and professional lives: How babies learn to see and identify objects
- Distinguishing, tracking, and naming: “Oh look, a **puppy** is coming!”, “This is a puppy”, “The puppy wants to kiss you”
  - Baby: “Hmhm... this furry, medium, four-legged, moving blob seems to be called **puppy**”
- This is a TRAINING process (*)

## SEEING LIKE A HUMAN, CONT.

- Modern computer vision systems are meant to emulate how human brains transform sensual stimuli into conceptual understanding
- The process allows computers to set their own set of **rules** to classify information based on TRAINING (*)

# SEEING LIKE A HUMAN, CONT.

- Modern computer vision systems are meant to emulate how human brains transform sensual stimuli into conceptual understanding
- The process allows computers to set their own set of **rules** to classify information based on TRAINING (*)




Credit: *Buduma (2017)*

- This process is called Convolutional Neural Network (or CNN)
- A set of “neurons” in charge of identifying unique bits of information...
- ...arranged in a network that allows for information sharing/processing...
- ... to eventually “tag” or “name” the input

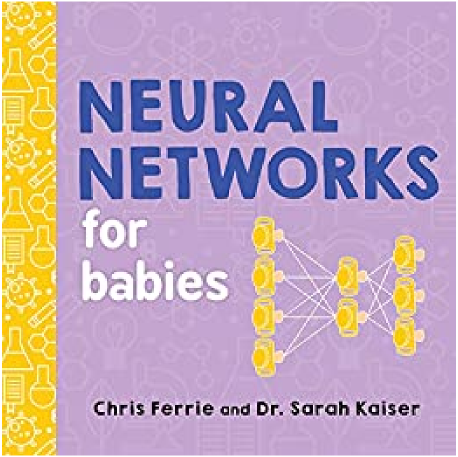
# WAIT... WHAT?

## WAIT... WHAT?

A very sophisticated text that I've been reading a lot recently:

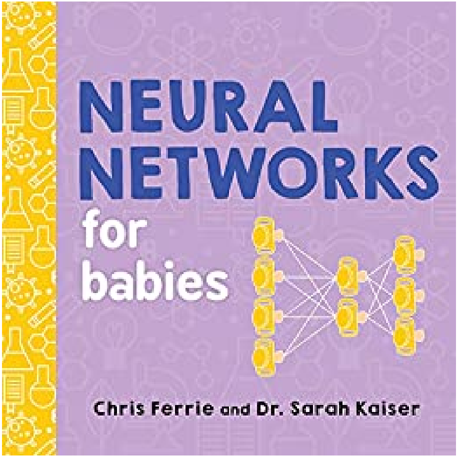
## WAIT… WHAT?

A very sophisticated text that I've been reading a lot recently:



## WAIT... WHAT?

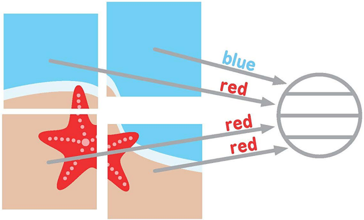
A very sophisticated text that I've been reading a lot recently:




(No... I am not joking)

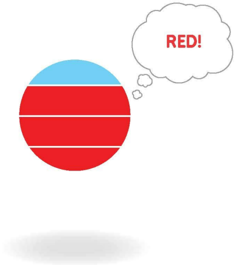
# The logic of CNNs

Is there a red animal in this picture?

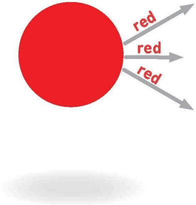



The neuron can decide based on its input.

# The logic of CNNs



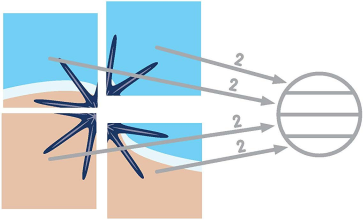

When the neuron has an answer,




it sends its own message.

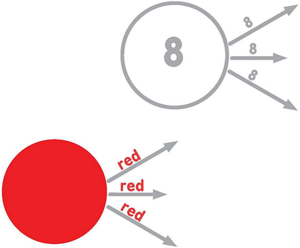
# The logic of CNNs

Does this animal have 8 arms?

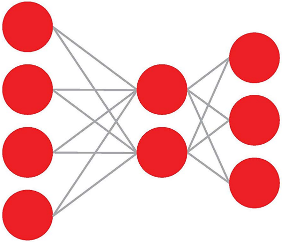



The neuron can decide based on its input.

# THE LOGIC OF CNNs




Where do the messages go?

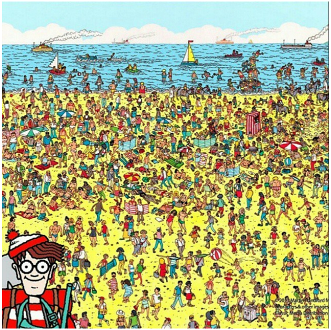


Neurons talk to each other.
They connect together in a network.

# Almost like finding Waldo…

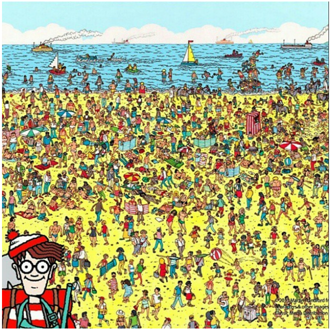
# ALMOST LIKE FINDING WALDO…

- Ok, not that cartoonish but almost!



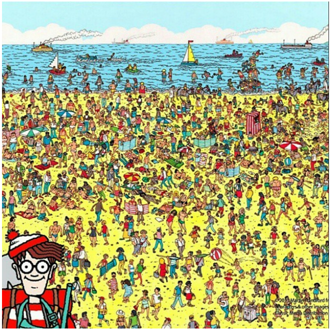
# ALMOST LIKE FINDING WALDO…

- Ok, not that cartoonish but almost!
- What is your approach when you want to find Waldo?



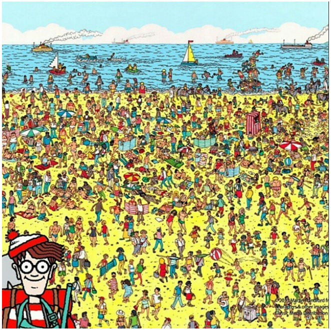
# ALMOST LIKE FINDING WALDO…

- Ok, not that cartoonish but almost!
- What is your approach when you want to find Waldo?
- Scan the image looking for particular "features"



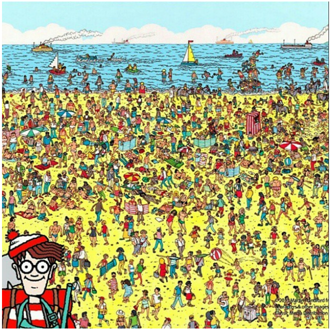
# ALMOST LIKE FINDING WALDO…

- Ok, not that cartoonish but almost!
- What is your approach when you want to find Waldo?
- Scan the image looking for particular "features"
  - Red and white stripes



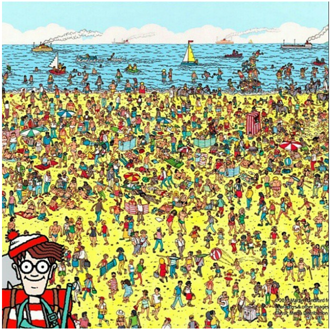
# Almost like finding Waldo…

- Ok, not that cartoonish but almost!
- What is your approach when you want to find Waldo?
- Scan the image looking for particular "features"
  - Red and white stripes
  - Glasses



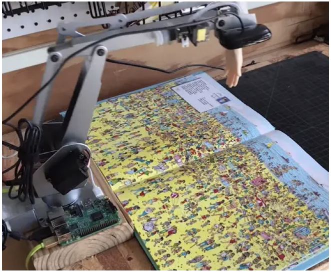
# ALMOST LIKE FINDING WALDO…

- Ok, not that cartoonish but almost!
- What is your approach when you want to find Waldo?
- Scan the image looking for particular "features"
    - Red and white stripes
    - Glasses
    - Hat



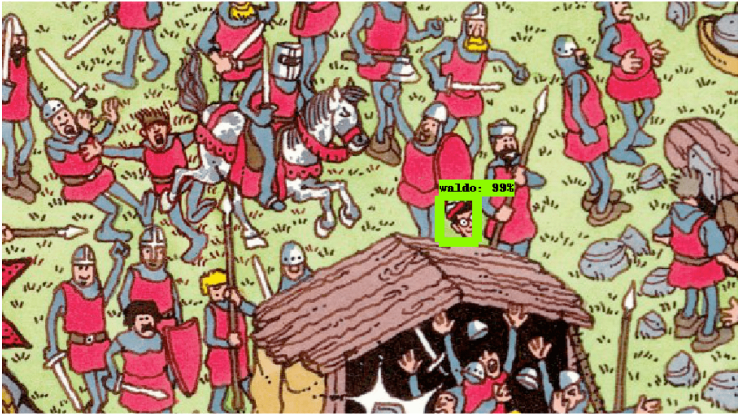
# Almost like finding Waldo...

- Ok, not that cartoonish but almost!
- What is your approach when you want to find Waldo?
- Scan the image looking for particular "features"
  - Red and white stripes
  - Glasses
  - Hat
- There is a robot who finds him in less than 5 seconds



# FOR REAL

And it's based on CNN code (see here)



# Speed bump: a healthy dose of skepticism

- Although the motivation behind and structure of CNNs try hard to resemble the human brain, they are EXTREMELY far from approaching its awesomeness

## Speed bump: a healthy dose of skepticism

- Although the motivation behind and structure of CNNs try hard to resemble the human brain, they are EXTREMELY far from approaching its awesomeness
- An obvious gap: context, meaning, abstraction

## Speed bump: a healthy dose of skepticism

- Although the motivation behind and structure of CNNs try hard to resemble the human brain, they are EXTREMELY far from approaching its awesomeness
- An obvious gap: context, meaning, abstraction
- An extra one: how large and rich the training set must be (ex. pumpkins and sheep)

## Speed bump: a healthy dose of skepticism

- Although the motivation behind and structure of CNNs try hard to resemble the human brain, they are EXTREMELY far from approaching its awesomeness
- An obvious gap: context, meaning, abstraction
- An extra one: how large and rich the training set must be (ex. pumpkins and sheep)
- This has important implications for their usage and applicability

Back to business...

# Convolutional Neural Networks (CNNs)

- Convolutional Neural Networks (CNNs) is a supervised learning algorithm to classify images

# Convolutional Neural Networks (CNNs)

- Convolutional Neural Networks (CNNs) is a supervised learning algorithm to classify images
- CNNs gradually learn what visual features of the image are more important in a classification task by transforming the image into multiple representations or *feature maps*.

# Convolutional Neural Networks (CNNs)

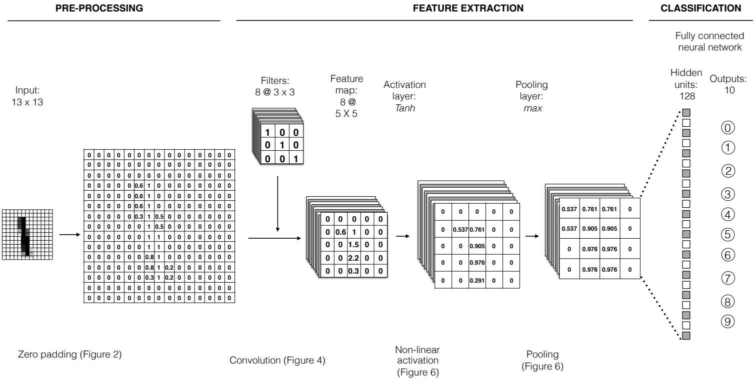
- Convolutional Neural Networks (CNNs) is a supervised learning algorithm to classify images
- CNNs gradually learn what visual features of the image are more important in a classification task by transforming the image into multiple representations or *feature maps*.
- CNNs are organized into multiple layers. Each layer contains multiple representations of the original image through maps of visual features such as edges, blobs or color combinations.

# Convolutional Neural Networks (CNNs)

- Convolutional Neural Networks (CNNs) is a supervised learning algorithm to classify images
- CNNs gradually learn what visual features of the image are more important in a classification task by transforming the image into multiple representations or *feature maps*.
- CNNs are organized into multiple layers. Each layer contains multiple representations of the original image through maps of visual features such as edges, blobs or color combinations.
- The part of learning and reaching a semantic concept that humans conduct by trial and error is achieved through the training, validation and testing procedures in CNNs.

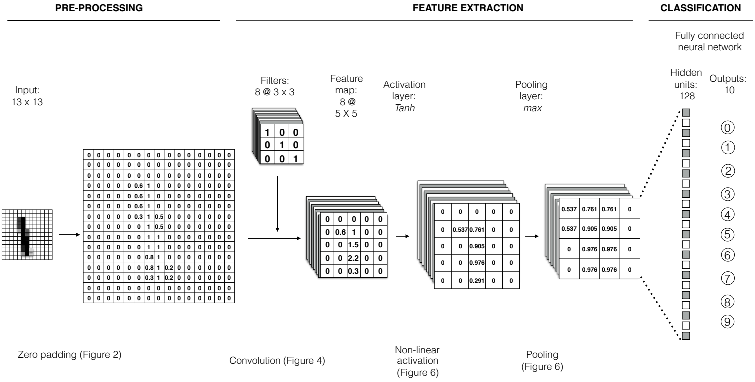
# NETWORK STRUCTURE

- GOAL: learn the features associated w/ outcomes



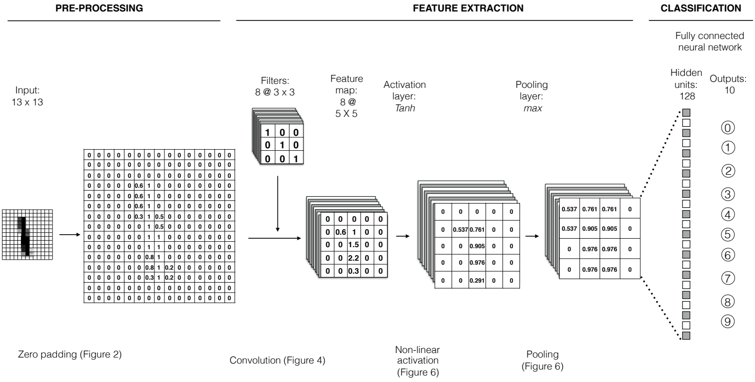
# Network structure

- GOAL: learn the features associated w/ outcomes
- Translation: obtain "coefficients" [weights in feature maps]



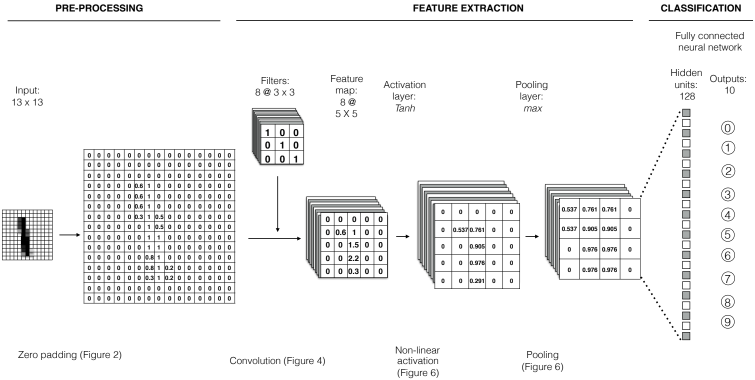
# NETWORK STRUCTURE

- GOAL: learn the features associated w/ outcomes
- Translation: obtain “coefficients” [weights in feature maps]
- Mainly, a data reduction technique → **Why?**

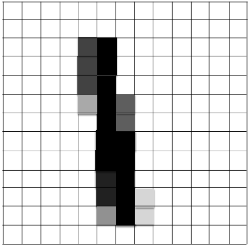


# NETWORK STRUCTURE

- GOAL: learn the features associated w/ outcomes
- Translation: obtain "coefficients" [weights in feature maps]
- Mainly, a data reduction technique → **Why?**
- Not a black-box! → Optimization of error



# REPRESENTING IMAGES

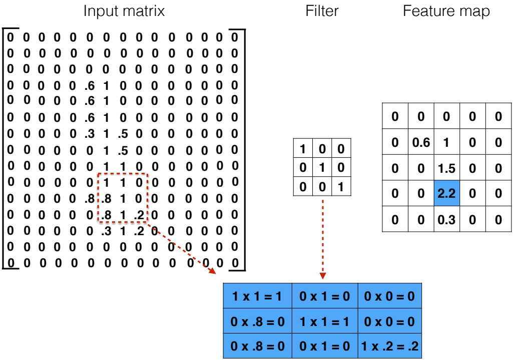


$$\begin{bmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&.6&1&0&0&0&0&0&0&0\\
0&0&0&0&.6&1&0&0&0&0&0&0&0\\
0&0&0&0&.6&1&0&0&0&0&0&0&0\\
0&0&0&0&.3&1&.5&0&0&0&0&0&0\\
0&0&0&0&0&1&.5&0&0&0&0&0&0\\
0&0&0&0&0&1&1&0&0&0&0&0&0\\
0&0&0&0&0&1&1&0&0&0&0&0&0\\
0&0&0&0&.8&.8&1&0&0&0&0&0&0\\
0&0&0&0&0&.8&1&.2&0&0&0&0&0\\
0&0&0&0&0&.3&1&.2&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0
\end{bmatrix}$$

The image is transformed into a numerical matrix, where each element represents the value of a specific pixel of the image measured as light intensity (in grayscale images) or color intensity (in color images).

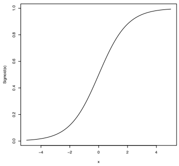
# FEATURE EXTRACTION

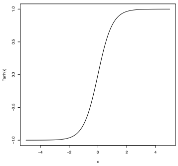
It's all about feature extraction!

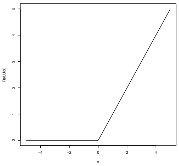



Filters are matrixes made of *weights*, that maximize or minimize the "intensity" of a pixel. Every filter slides through each 3 x 3 pixel area of the image, and computes the dot product of the region. The result is recorded on a smaller matrix to create *feature maps*. Intuitively, we want to detect whether and where a feature represented by a filter is prominent in the image.

# ACTIVATION FUNCTIONS








(a) $Sigmoid(x) = \frac{1}{1+e^{-x}}$ (b) $Tanh(x) = \frac{2}{1+e^{-2x}}$ (c)$ReLU(x) = \begin{cases} 0 & \text{if } x < 0, \\ x & \text{otherwise.} \end{cases}$

We add non-linearity by including an *activation layer*.

# POOLING STAGE

*Non-linear activation:*
*Tanh(x)*

*max pooling*

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0.6 | 1 | 0 | 0 |
| 0 | 0 | 1.5 | 0 | 0 |
| 0 | 0 | 2.2 | 0 | 0 |
| 0 | 0 | 0.3 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0.537 | 0.761 | 0 | 0 |
| 0 | 0 | 0.905 | 0 | 0 |
| 0 | 0 | 0.976 | 0 | 0 |
| 0 | 0 | 0.291 | 0 | 0 |

| | | | |
|---|---|---|---|
| 0.537 | 0.761 | 0.761 | 0 |
| 0.537 | 0.905 | 0.905 | 0 |
| 0 | 0.976 | 0.976 | 0 |
| 0 | 0.976 | 0.976 | 0 |

Once the activation map shows non-linear outputs, we reduce its dimensionality using a *pooling layer*. A pooling layer shrinks the size of the matrix while keeping the most important information in the feature map.

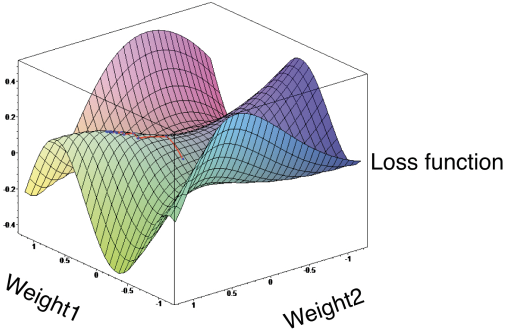
# LEARNING

- The last stage of the network involves the classification of the image. The way in which the CNN learns the features that correlate to each outcome follows a procedure called back-propagation.

More on back-propagation

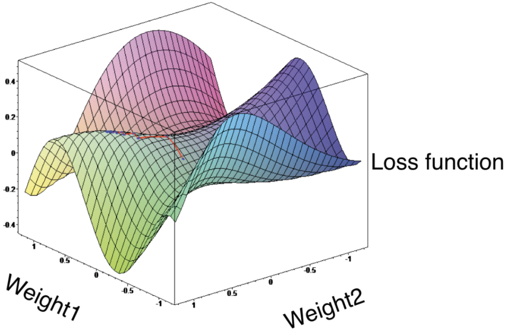
# ACTUALLY, THIS SHOULD BE FAMILIAR…

## Loss function



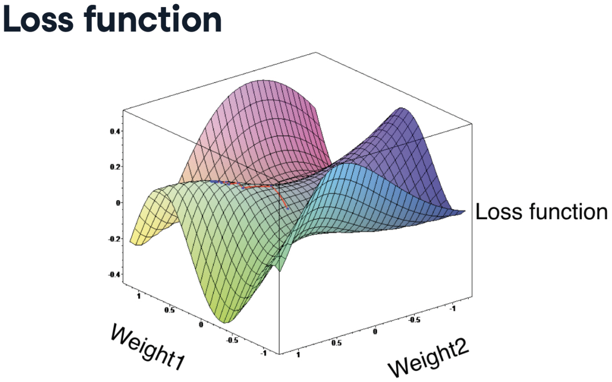
# ACTUALLY, THIS SHOULD BE FAMILIAR…

## Loss function



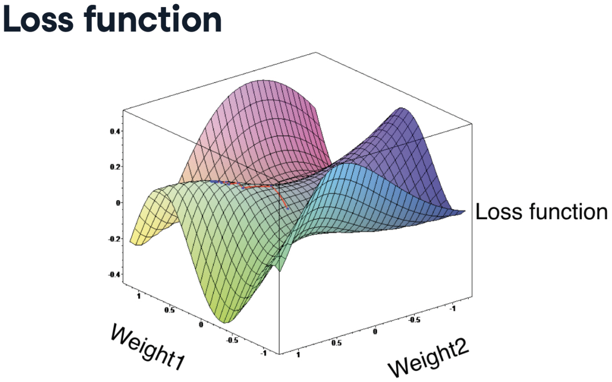

- Minimize multidimensional loss function →

# ACTUALLY, THIS SHOULD BE FAMILIAR...




- Minimize multidimensional loss function →

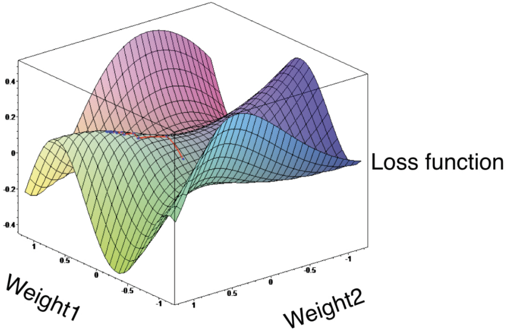
# ACTUALLY, THIS SHOULD BE FAMILIAR…




- Minimize multidimensional loss function → (OLS anyone?)
- By finding the minimum point [=minimum prediction error]

# ACTUALLY, THIS SHOULD BE FAMILIAR...

## Loss function

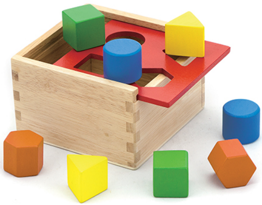



- Minimize multidimensional loss function → (OLS anyone?)
- By finding the minimum point [=minimum prediction error]
- Explore the “field” step by step

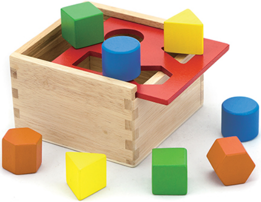
# BEYOND A CNN: TRANSFER LEARNING

# BEYOND A CNN: TRANSFER LEARNING

# Beyond a CNN: transfer learning



# Beyond a CNN: transfer learning



# BEYOND A CNN: TRANSFER LEARNING (CONT.)

## BEYOND A CNN: TRANSFER LEARNING (CONT.)

- Why *transfer learning*? Training a CNN from scratch is expensive and intensive! Always remember the pumpkins and sheep examples

# Beyond a CNN: transfer learning (cont.)

- Why *transfer learning*? Training a CNN from scratch is expensive and intensive! Always remember the pumpkins and sheep examples
- Core idea:

# BEYOND A CNN: TRANSFER LEARNING (CONT.)

- Why *transfer learning*? Training a CNN from scratch is expensive and intensive! Always remember the pumpkins and sheep examples
- Core idea:
  1. Take advantage of features that have already been learned

## Beyond a CNN: transfer learning (cont.)

- Why *transfer learning*? Training a CNN from scratch is expensive and intensive! Always remember the pumpkins and sheep examples
- Core idea:
  1. Take advantage of features that have already been learned
  2. Learn new things *on top* of the old knowledge from new experiences

## BEYOND A CNN: TRANSFER LEARNING (CONT.)

- Why *transfer learning*? Training a CNN from scratch is expensive and intensive! Always remember the pumpkins and sheep examples
- Core idea:
  1. Take advantage of features that have already been learned
  2. Learn new things *on top* of the old knowledge from new experiences
- CNN world:

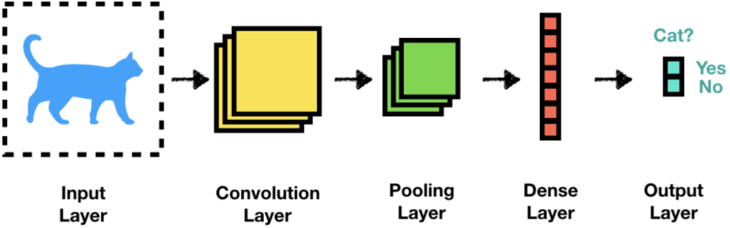
# Beyond a CNN: transfer learning (cont.)

- Why *transfer learning*? Training a CNN from scratch is expensive and intensive! Always remember the pumpkins and sheep examples
- Core idea:
  1. Take advantage of features that have already been learned
  2. Learn new things *on top* of the old knowledge from new experiences
- CNN world:
  1. Keep information processed by a given layer of a CNN

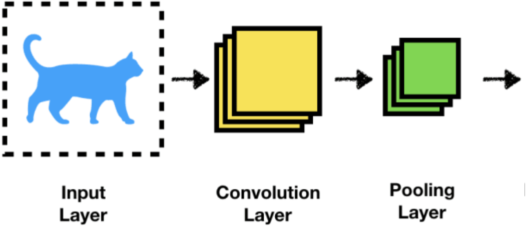
# Beyond a CNN: transfer learning (cont.)

- Why *transfer learning*? Training a CNN from scratch is expensive and intensive! Always remember the pumpkins and sheep examples
- Core idea:
  1. Take advantage of features that have already been learned
  2. Learn new things *on top* of the old knowledge from new experiences
- CNN world:
  1. Keep information processed by a given layer of a CNN
  2. Retrain further layers with new data and new labels

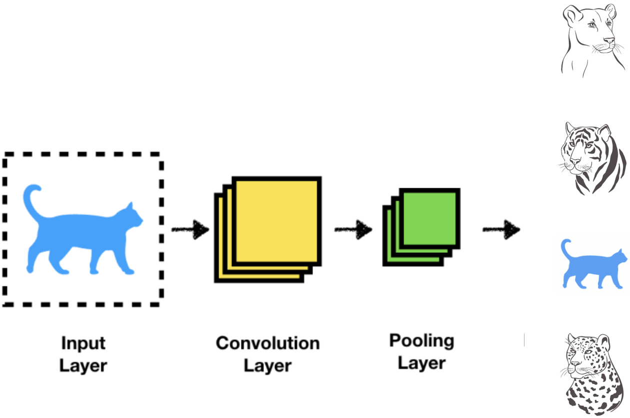
# BEYOND A CNN: TRANSFER LEARNING (CONT.)



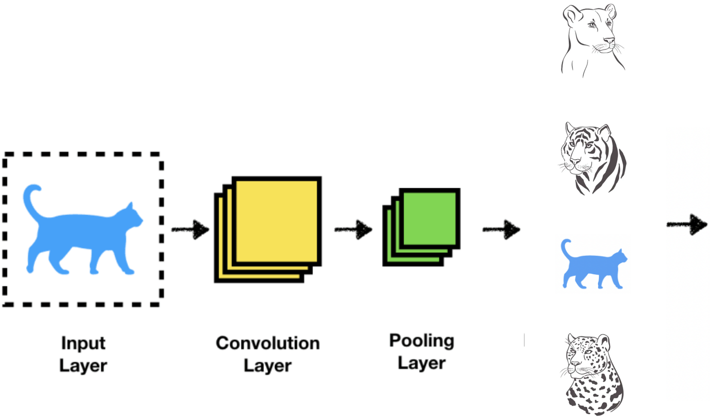
# BEYOND A CNN: TRANSFER LEARNING (CONT.)



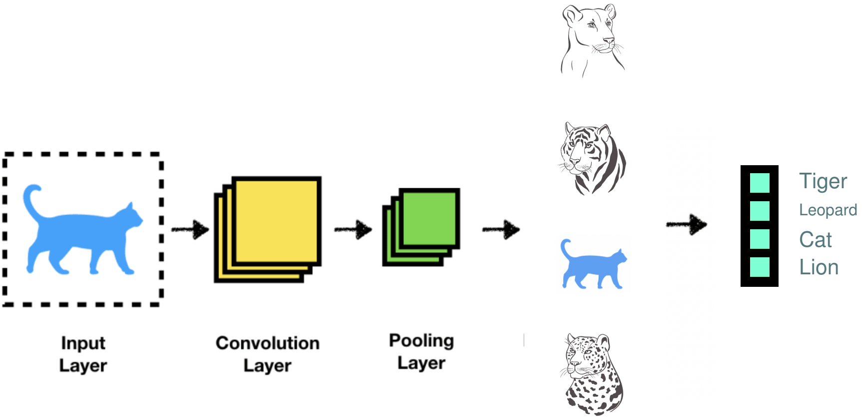
# BEYOND A CNN: TRANSFER LEARNING (CONT.)



# BEYOND A CNN: TRANSFER LEARNING (CONT.)

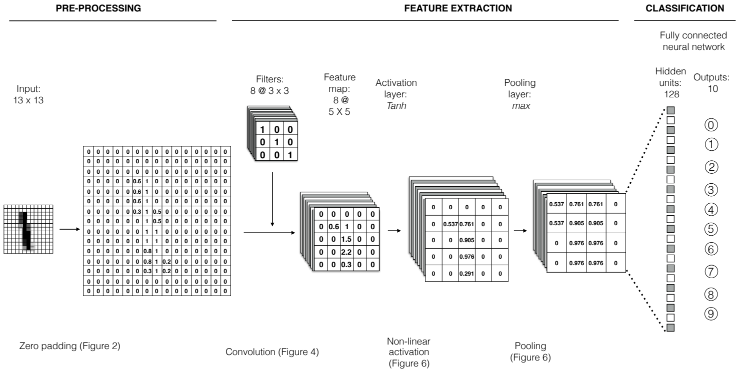


# Beyond a CNN: transfer learning (cont.)



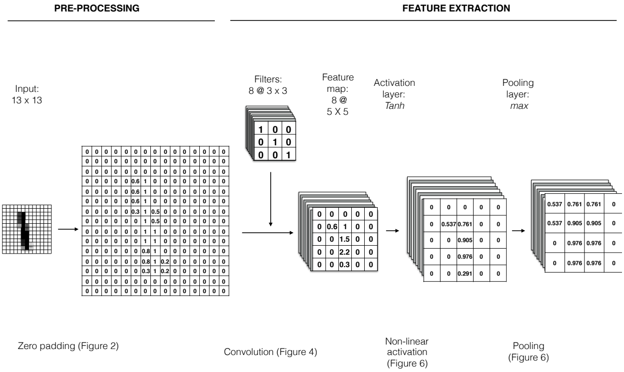
# BEYOND A CNN: TRANSFER LEARNING (CONT.)

- “Freeze” some layers and retrain the active ones
- Idea: keep useful learned features and fine-tune to account for your labels of interest



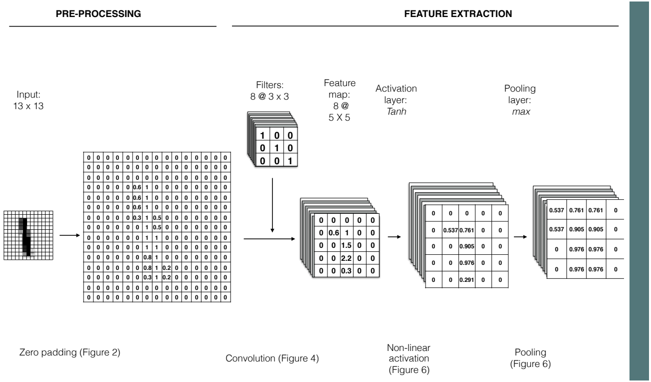
# BEYOND A CNN: TRANSFER LEARNING (CONT.)

- “Freeze” some layers and retrain the active ones
- Idea: keep useful learned features and fine-tune to account for your labels of interest



# BEYOND A CNN: TRANSFER LEARNING (CONT.)

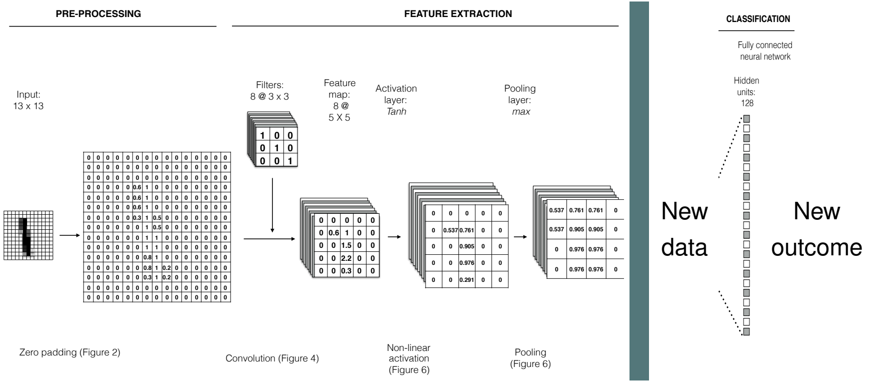
- “Freeze” some layers and retrain the active ones
- Idea: keep useful learned features and fine-tune to account for your labels of interest




New data

# Beyond a CNN: Transfer Learning (cont.)

- "Freeze" some layers and retrain the active ones
- Idea: keep useful learned features and fine-tune to account for your labels of interest



## CHALLENGES AND RECOMMENDATIONS

- Prevent overfitting(*)
  - Increase number of training images
  - Data augmentation
  - Dropout random neurons
- Optimize your training set
  - Active learning: Informativeness vs. Representativeness
  - Class balance
  - "Denoise" images
  - Batch normalization
  - CAUTION: Bias training
- Post-CNN diagnosis
  - Know your training, testing and out-of-sample data
  - Always check mislabeled examples: validate, validate, validate...
  - Diagnosis
  - Hyperparameter grid for tuning

# Going beyond prediction: Beyond a CNN

- There is more than just labeling and prediction

# Going beyond prediction: Beyond a CNN

- There is more than just labeling and prediction
- Maybe we want to:

# Going beyond prediction: Beyond a CNN

- There is more than just labeling and prediction
- Maybe we want to:
    - Cluster images based on content similarities

## Going beyond prediction: Beyond a CNN

- There is more than just labeling and prediction
- Maybe we want to:
  - Cluster images based on content similarities
  - Use visual features to build a scale

## Going beyond prediction: Beyond a CNN

- There is more than just labeling and prediction
- Maybe we want to:
  - Cluster images based on content similarities
  - Use visual features to build a scale
  - Identify and measure the proportion of *k* topics in each image

## Going beyond prediction: Beyond a CNN

- There is more than just labeling and prediction
- Maybe we want to:
    - Cluster images based on content similarities
    - Use visual features to build a scale
    - Identify and measure the proportion of *k* topics in each image
    - See the correlation between visual features and predictions (regression much?)

# Going beyond prediction: Beyond a CNN

- There is more than just labeling and prediction
- Maybe we want to:
  - Cluster images based on content similarities
  - Use visual features to build a scale
  - Identify and measure the proportion of *k* topics in each image
  - See the correlation between visual features and predictions (regression much?)
  - Inspect and visualize features (*)

# Going beyond prediction: Beyond a CNN

- There is more than just labeling and prediction
- Maybe we want to:
  - Cluster images based on content similarities
  - Use visual features to build a scale
  - Identify and measure the proportion of *k* topics in each image
  - See the correlation between visual features and predictions (regression much?)
  - Inspect and visualize features (*)
  - ...

# Going beyond prediction: Beyond a CNN

- There is more than just labeling and prediction
- Maybe we want to:
  - Cluster images based on content similarities
  - Use visual features to build a scale
  - Identify and measure the proportion of *k* topics in each image
  - See the correlation between visual features and predictions (regression much?)
  - Inspect and visualize features (*)
  - ...
- If so, we might need other tools and approaches

# The Bag of Visual Words

# What we are going to build today

## What we are going to build today

1. Follow a Bag of Visuals Word approach in text BUT with images

## What we are going to build today

1. Follow a Bag of Visuals Word approach in text BUT with images
2. Build a Document-Term Matrix with images $\Rightarrow$ Image-Visual Word Matrix

## What we are going to build today

1. Follow a Bag of Visuals Word approach in text BUT with images
2. Build a Document-Term Matrix with images $\Rightarrow$ Image-Visual Word Matrix
   - Learn to tokenize images (useful for unsupervised and interpretable methods)

## WHAT WE ARE GOING TO BUILD TODAY

1. Follow a Bag of Visuals Word approach in text BUT with images
2. Build a Document-Term Matrix with images $\Rightarrow$ Image-Visual Word Matrix
    - Learn to tokenize images (useful for unsupervised and interpretable methods)
    - Bring to the plate what we learned about CNNs

## What we are going to build today

1. Follow a Bag of Visuals Word approach in text BUT with images
2. Build a Document-Term Matrix with images $\Rightarrow$ Image-Visual Word Matrix
   - Learn to tokenize images (useful for unsupervised and interpretable methods)
   - Bring to the plate what we learned about CNNs
3. And then...?

## What we are going to build today

1. Follow a Bag of Visuals Word approach in text BUT with images
2. Build a Document-Term Matrix with images $\Rightarrow$ Image-Visual Word Matrix
   - Learn to tokenize images (useful for unsupervised and interpretable methods)
   - Bring to the plate what we learned about CNNs
3. And then...? Choose your method! (Today I chose for you)

# WHAT WE ARE GOING TO BUILD TODAY

1. Follow a Bag of Visuals Word approach in text BUT with images
2. Build a Document-Term Matrix with images ⇒ Image-Visual Word Matrix
   - Learn to tokenize images (useful for unsupervised and interpretable methods)
   - Bring to the plate what we learned about CNNs
3. And then...? Choose your method! (Today I chose for you)

Latent Treatment Identification

## What we are going to build today

1. Follow a Bag of Visuals Word approach in text BUT with images
2. Build a Document-Term Matrix with images ⇒ Image-Visual Word Matrix
   - Learn to tokenize images (useful for unsupervised and interpretable methods)
   - Bring to the plate what we learned about CNNs
3. And then...? Choose your method! (Today I chose for you)

Latent Treatment Identification

Binary Classification

## What we are going to build today

1. Follow a Bag of Visuals Word approach in text BUT with images
2. Build a Document-Term Matrix with images $\Rightarrow$ Image-Visual Word Matrix
   - Learn to tokenize images (useful for unsupervised and interpretable methods)
   - Bring to the plate what we learned about CNNs
3. And then...? Choose your method! (Today I chose for you)

Latent Treatment Identification

Binary Classification

Clustering Analysis

## What we are going to build today

1. Follow a Bag of Visuals Word approach in text BUT with images
2. Build a Document-Term Matrix with images ⇒ Image-Visual Word Matrix
   - Learn to tokenize images (useful for unsupervised and interpretable methods)
   - Bring to the plate what we learned about CNNs
3. And then...? Choose your method! (Today I chose for you)

Latent Treatment Identification | Binary Classification | Clustering Analysis | Topic Modeling

# WHAT WE ARE GOING TO BUILD TODAY

1. Follow a Bag of Visuals Word approach in text BUT with images
2. Build a Document-Term Matrix with images ⇒ Image-Visual Word Matrix
   - Learn to tokenize images (useful for unsupervised and interpretable methods)
   - Bring to the plate what we learned about CNNs
3. And then...? Choose your method! (Today I chose for you)

Topic Modeling

## Today: A "Visual" Structural Topic Model

- Structural Topic Model (Roberts et al. 2014)

## Today: A "Visual" Structural Topic Model

- Structural Topic Model (Roberts et al. 2014)
- Tool for topic modeling of texts with document-level covariate information

# Today: A "Visual" Structural Topic Model

- Structural Topic Model (Roberts et al. 2014)
- Tool for topic modeling of texts with document-level covariate information
- Mixture model:

# Today: A "Visual" Structural Topic Model

- Structural Topic Model (Roberts et al. 2014)
- Tool for topic modeling of texts with document-level covariate information
- Mixture model:
  - Probability that words belong to each of the "topics" or groups of interest

# Today: A "Visual" Structural Topic Model

- Structural Topic Model (Roberts et al. 2014)
- Tool for topic modeling of texts with document-level covariate information
- Mixture model:
  - Probability that words belong to each of the "topics" or groups of interest
  - Not a single classification outcome, but proportions of all potential topics for each document

# Today: A "Visual" Structural Topic Model

- Structural Topic Model (Roberts et al. 2014)
- Tool for topic modeling of texts with document-level covariate information
- Mixture model:
  - Probability that words belong to each of the "topics" or groups of interest
  - Not a single classification outcome, but proportions of all potential topics for each document

# TODAY: A "VISUAL" STRUCTURAL TOPIC MODEL

- Structural Topic Model (Roberts et al. 2014)
- Tool for topic modeling of texts with document-level covariate information
- Mixture model:
  - Probability that words belong to each of the "topics" or groups of interest
  - Not a single classification outcome, but proportions of all potential topics for each document

Sky:

# Today: A "Visual" Structural Topic Model

- Structural Topic Model (Roberts et al. 2014)
- Tool for topic modeling of texts with document-level covariate information
- Mixture model:
  - Probability that words belong to each of the "topics" or groups of interest
  - Not a single classification outcome, but proportions of all potential topics for each document

Sky:

Crowd:

# Today: A "Visual" Structural Topic Model

- Structural Topic Model (Roberts et al. 2014)
- Tool for topic modeling of texts with document-level covariate information
- Mixture model:
  - Probability that words belong to each of the "topics" or groups of interest
  - Not a single classification outcome, but proportions of all potential topics for each document

Sky:

Crowd:

Pavement:

# TODAY: A "VISUAL" STRUCTURAL TOPIC MODEL

- Structural Topic Model (Roberts et al. 2014)
- Tool for topic modeling of texts with document-level covariate information
- Mixture model:
  - Probability that words belong to each of the "topics" or groups of interest
  - Not a single classification outcome, but proportions of all potential topics for each document

Sky:

Crowd:

Pavement:

Flag:

# TODAY: A "VISUAL" STRUCTURAL TOPIC MODEL

- Structural Topic Model (Roberts et al. 2014)
- Tool for topic modeling of texts with document-level covariate information
- Mixture model:
  - Probability that words belong to each of the "topics" or groups of interest
  - Not a single classification outcome, but proportions of all potential topics for each document

Sky: 41%

Crowd: 38%

Pavement: 12%

Flag: 9%

## BUT FIRST: CONSTRUCTING VISUAL WORDS

- Why do we need visual words?

# But first: Constructing Visual Words

- Why do we need visual words?
- To build a Document-Term matrix (DTM)!

## But first: Constructing Visual Words

- Why do we need visual words?
- To build a Document-Term matrix (DTM)!
- Why a DTM?

## But first: Constructing Visual Words

- Why do we need visual words?
- To build a Document-Term matrix (DTM)!
- Why a DTM?
- Because that's the input of a STM

## But first: Constructing Visual Words

- Why do we need visual words?
- To build a Document-Term matrix (DTM)!
- Why a DTM?
- Because that's the input of a STM
- Actually, what's a DTM?

# But first: Constructing Visual Words, cont.

# But first: Constructing Visual Words, cont.

1. Identification of blocks in images

## But first: Constructing Visual Words, cont.

1. Identification of blocks in images
2. Extraction of features using a CNN

## But first: Constructing Visual Words, cont.

1. Identification of blocks in images
2. Extraction of features using a CNN
3. Construction of visual vocabulary based on clustering features

# But first: Constructing Visual Words, cont.

1. Identification of blocks in images
2. Extraction of features using a CNN
3. Construction of visual vocabulary based on clustering features
4. Construction of Image-Visual Word matrix

# DIVISION OF IMAGES INTO BLOCKS


**(a)** Original image (resized)

# Division of Images into Blocks


**(a)** Original image (resized)

**(b)** Image divided into 32 × 32 pixels blocks

# Feature Extraction with CNNs

# Feature Extraction with CNNs

- Use a **CNN** to extract features from EACH of the “mini” images composing each of the images in our corpus

# Feature Extraction with CNNs

- Use a **CNN** to extract features from EACH of the “mini” images composing each of the images in our corpus
- CNN = Convolutional Neural Network

# FEATURE EXTRACTION USING CNNS

# Feature extraction using CNNs

- Use pre-trained model on each block of an image

# Feature extraction using CNNs

- Use pre-trained model on each block of an image
- The CNN creates feature maps of “elements/descriptors” that can be found in an image

# Feature extraction using CNNs

- Use pre-trained model on each block of an image
- The CNN creates feature maps of “elements/descriptors” that can be found in an image
- Remove the dense layer (the final one) and keep an appropriate feature map vector → “Predictors”

# Feature extraction using CNNs

- Use pre-trained model on each block of an image
- The CNN creates feature maps of “elements/descriptors” that can be found in an image
- Remove the dense layer (the final one) and keep an appropriate feature map vector → “Predictors”
- = Each image is described by vector of size *number of blocks* × *number of features from CNN*

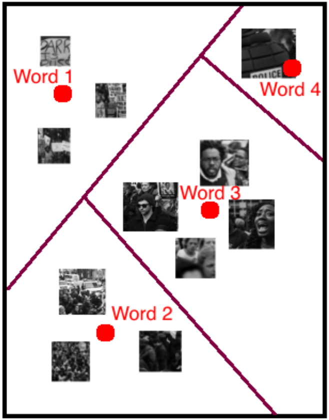
# Feature extraction using CNNs

- Use pre-trained model on each block of an image
- The CNN creates feature maps of “elements/descriptors” that can be found in an image
- Remove the dense layer (the final one) and keep an appropriate feature map vector → “Predictors”
- = Each image is described by vector of size *number of blocks* × *number of features from CNN*
- In our applications, this is 70×2,048

# Clustering Features to Build Visual Vocabulary

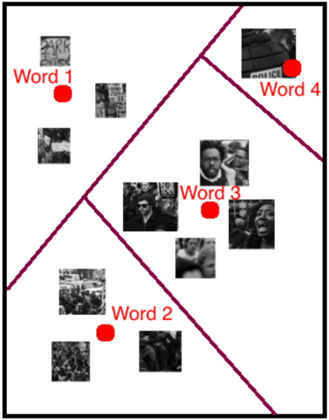
# Clustering Features to Build Visual Vocabulary

- Need for tokens → Words in columns of a DTM



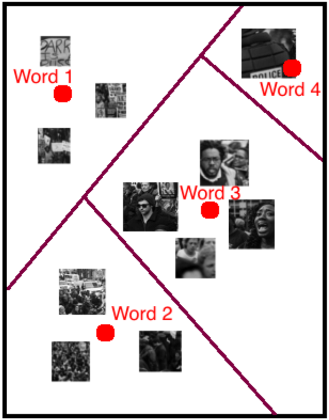
# Clustering Features to Build Visual Vocabulary

- Need for tokens → Words in columns of a DTM
- Define *v* clusters (= # of desired visual words)



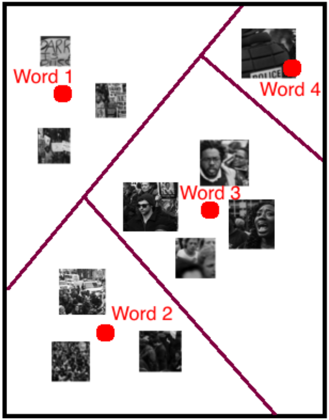
# Clustering Features to Build Visual Vocabulary

- Need for tokens → Words in columns of a DTM
- Define *v* clusters (= # of desired visual words)
- Cluster randomly selected sample of feature vectors



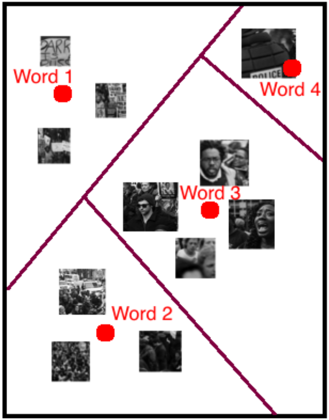
# Clustering Features to Build Visual Vocabulary

- Need for tokens → Words in columns of a DTM
- Define *v* clusters (= # of desired visual words)
- Cluster randomly selected sample of feature vectors
- Centroid of cluster is the "visual word"



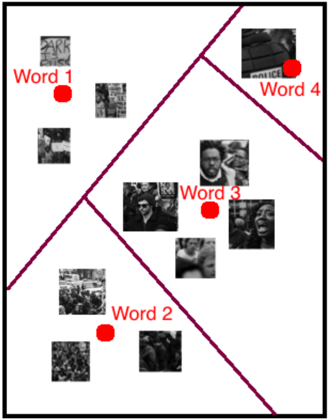
# Clustering Features to Build Visual Vocabulary

- Need for tokens → Words in columns of a DTM
- Define *v* clusters (= # of desired visual words)
- Cluster randomly selected sample of feature vectors
- Centroid of cluster is the "visual word"
- Why do we do this?



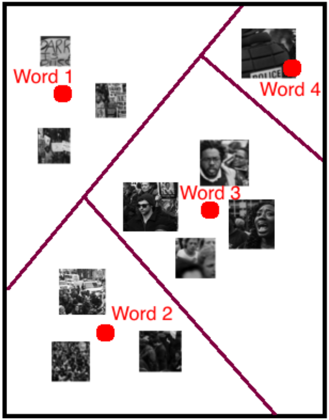
# Clustering Features to Build Visual Vocabulary

- Need for tokens → Words in columns of a DTM
- Define *v* clusters (= # of desired visual words)
- Cluster randomly selected sample of feature vectors
- Centroid of cluster is the "visual word"
- Why do we do this?
  - Similar features = Same concept



# Clustering Features to Build Visual Vocabulary

- Need for tokens → Words in columns of a DTM
- Define *v* clusters (= # of desired visual words)
- Cluster randomly selected sample of feature vectors
- Centroid of cluster is the "visual word"
- Why do we do this?
  - Similar features = Same concept
  - Reduce potential sparsity in IVWM



# Visualizing Visual Words

- Blocks that belong to a given cluster are similar in terms of feature vectors

## Visualizing Visual Words

- Blocks that belong to a given cluster are similar in terms of feature vectors
- Should look visually similar

# Visualizing Visual Words

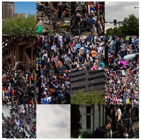
- Blocks that belong to a given cluster are similar in terms of feature vectors
- Should look visually similar
- Construct "visual words" using the 16 feature vectors closest to each of the centroid of the cluster

# Visualizing Visual Words

- Blocks that belong to a given cluster are similar in terms of feature vectors
- Should look visually similar
- Construct “visual words” using the 16 feature vectors closest to each of the centroid of the cluster
- E.g. the most similar blocks to the “average” block representing the cluster

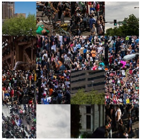
# Visualizing Visual Words

- Blocks that belong to a given cluster are similar in terms of feature vectors
- Should look visually similar
- Construct "visual words" using the 16 feature vectors closest to each of the centroid of the cluster
- E.g. the most similar blocks to the "average" block representing the cluster



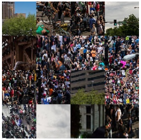
# Visualizing Visual Words

- Blocks that belong to a given cluster are similar in terms of feature vectors
- Should look visually similar
- Construct "visual words" using the 16 feature vectors closest to each of the centroid of the cluster
- E.g. the most similar blocks to the "average" block representing the cluster



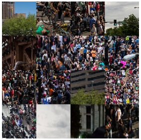
# Visualizing Visual Words

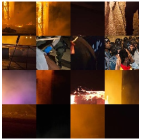
- Blocks that belong to a given cluster are similar in terms of feature vectors
- Should look visually similar
- Construct “visual words” using the 16 feature vectors closest to each of the centroid of the cluster
- E.g. the most similar blocks to the “average” block representing the cluster



# Visualizing Visual Words

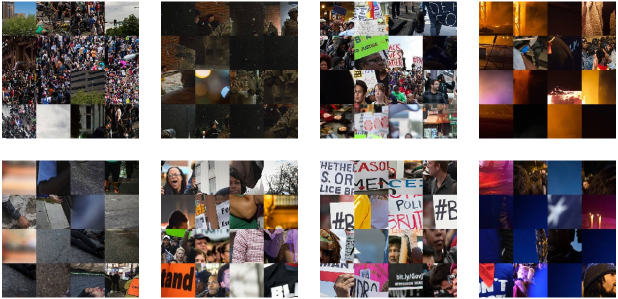
- Blocks that belong to a given cluster are similar in terms of feature vectors
- Should look visually similar
- Construct “visual words” using the 16 feature vectors closest to each of the centroid of the cluster
- E.g. the most similar blocks to the “average” block representing the cluster



# Visualizing Visual Words

- Blocks that belong to a given cluster are similar in terms of feature vectors
- Should look visually similar
- Construct “visual words” using the 16 feature vectors closest to each of the centroid of the cluster
- E.g. the most similar blocks to the “average” block representing the cluster



# Image-Visual Word Matrix

| Document/Term | Black | Lives | ⋯ | Matter | Ferguson | protest |
|---|---|---|---|---|---|---|
| Where was this public display of support during the Black Lives Matter movement or the prolonged demonstrations in Ferguson? | 1 | 1 | ⋯ | 1 | 1 | 0 |
| And to be honest with you, we wouldn't be seeing this level of protest if we didn't have this for the last five years. Black Lives Matter really set this idea of how we fight and how we protest into action. | 1 | 1 | ⋯ | 1 | 0 | 2 |
| Over the past several weeks, the students of Marjory Stoneman Douglas High School, have seized the national spotlight and joined a proud tradition of student-led protest movements. | 0 | 0 | ⋯ | 0 | 0 | 1 |

# Image-Visual Word Matrix (cont.)

| Image/Visual Word |  |  | … |  |
|---|---|---|---|---|
|  | 0 | 1 | … | 0 |
|  | 0 | 1 | … | 1 |
|  | 5 | 8 | … | 4 |

# Building the IVWM to emulate DTM

## Building the IVWM to emulate DTM

Count the number of times each visual word appears in an image

## Building the IVWM to emulate DTM

Count the number of times each visual word appears in an image

- Also not trivial...

# Building the IVWM to emulate DTM

Count the number of times each visual word appears in an image

- Also not trivial...
- Assign each feature vector to the most similar visual word in the vocabulary

# Building the IVWM to emulate DTM

Count the number of times each visual word appears in an image

- Also not trivial...
- Assign each feature vector to the most similar visual word in the vocabulary
  - Compute the Euclidean distance between each feature vector and the centroids of the clusters

## BUILDING THE IVWM TO EMULATE DTM

Count the number of times each visual word appears in an image

- Also not trivial...
- Assign each feature vector to the most similar visual word in the vocabulary
  - Compute the Euclidean distance between each feature vector and the centroids of the clusters
  - Assign feature vector to visual word with shortest distance to centroid

# Illustrating example: Migrant Caravan

# Illustrating example: Migrant Caravan

- Groups of migrants from Central America fleeing violence in their countries and seeking refugee in the U.S.

# Illustrating example: Migrant Caravan

- Groups of migrants from Central America fleeing violence in their countries and seeking refugee in the U.S.
- Very polarized coverage of this phenomenon

## Illustrating example: Migrant Caravan

- Groups of migrants from Central America fleeing violence in their countries and seeking refugee in the U.S.
- Very polarized coverage of this phenomenon
- Emphasis on magnitude: threat, invasion

# Illustrating example: Migrant Caravan

- Groups of migrants from Central America fleeing violence in their countries and seeking refugee in the U.S.
- Very polarized coverage of this phenomenon
- Emphasis on magnitude: threat, invasion

"Massive migrant caravan on the way"

"Looks more like an invasion than anything"

# Illustrating example: Migrant Caravan

- Groups of migrants from Central America fleeing violence in their countries and seeking refugee in the U.S.
- Very polarized coverage of this phenomenon
- Emphasis on magnitude: threat, invasion

"Massive migrant caravan on the way"

"Looks more like an invasion than anything"


"See them as they are: Desperate, leaving behind whatever they had, and whomever they knew, all for a better chance at life"

## Identifying political components in the data generation process of images

- Goal: Identify and quantify the visual framing of the magnitude of the caravan

## Identifying political components in the data generation process of images

- Goal: Identify and quantify the visual framing of the magnitude of the caravan
- Structural Topic Model to identify underlying "topics", understood as frames, in the images

# Identifying political components in the data generation process of images

- Goal: Identify and quantify the visual framing of the magnitude of the caravan
- Structural Topic Model to identify underlying “topics”, understood as frames, in the images
- Visual codebook generated from $\approx$ 6,000 pictures from *Getty*

# Identifying political components in the data generation process of images

- Goal: Identify and quantify the visual framing of the magnitude of the caravan
- Structural Topic Model to identify underlying “topics”, understood as frames, in the images
- Visual codebook generated from $\approx$ 6,000 pictures from *Getty*
- 500 words vocabulary

# Identifying political components in the data generation process of images

- Goal: Identify and quantify the visual framing of the magnitude of the caravan
- Structural Topic Model to identify underlying “topics”, understood as frames, in the images
- Visual codebook generated from ≈ 6,000 pictures from *Getty*
- 500 words vocabulary
- Prevalence covariates: agency and date

## Identifying political components in the data generation process of images

- Goal: Identify and quantify the visual framing of the magnitude of the caravan
- Structural Topic Model to identify underlying "topics", understood as frames, in the images
- Visual codebook generated from $\approx$ 6,000 pictures from *Getty*
- 500 words vocabulary
- Prevalence covariates: agency and date
- Selection of 6 topics:

# Identifying political components in the data generation process of images

- Goal: Identify and quantify the visual framing of the magnitude of the caravan
- Structural Topic Model to identify underlying "topics", understood as frames, in the images
- Visual codebook generated from $\approx$ 6,000 pictures from *Getty*
- 500 words vocabulary
- Prevalence covariates: agency and date
- Selection of 6 topics:
  - Crowd

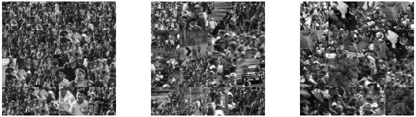
# Identifying political components in the data generation process of images

- Goal: Identify and quantify the visual framing of the magnitude of the caravan
- Structural Topic Model to identify underlying “topics”, understood as frames, in the images
- Visual codebook generated from $\approx$ 6,000 pictures from *Getty*
- 500 words vocabulary
- Prevalence covariates: agency and date
- Selection of 6 topics:
  - Crowd
  - Border/Fence, Small group/Portrait, Water/Sky, Camps, Darkness

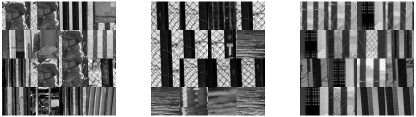
## Underlying topics in the caravan: FREX words

# Underlying topics in the caravan: FREX words

**Topic 1**: Crowds



**Topic 2**: Border/Fence



**Topic 3**: Water/Sky

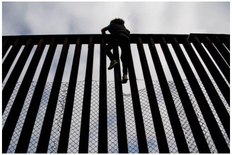
# Underlying topics in the caravan: representative images

# Underlying topics in the caravan: representative images

Crowd

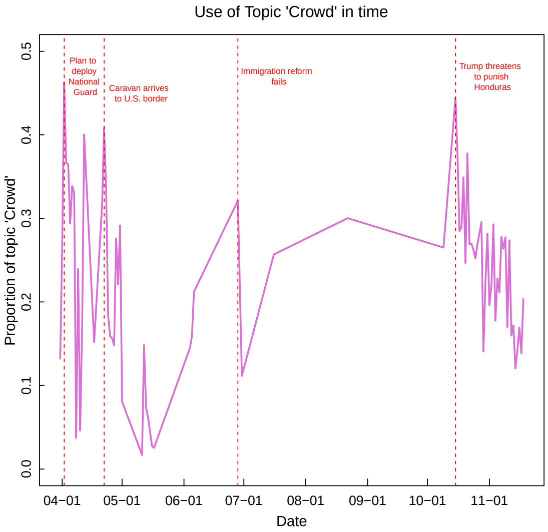


Border/ Fence



Water/ Sky


# CROWD TOPIC IN TIME



# Validation: high correlation between topics and manual coding

## Validation: high correlation between topics and manual coding

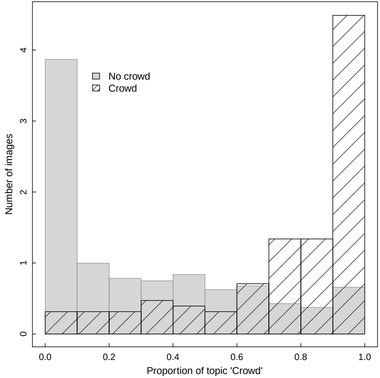
- Hand-coded sample: presence of medium/big crowd in the image (crowd=1) or no (crowd=0).

## Validation: high correlation between topics and manual coding

- Hand-coded sample: presence of medium/big crowd in the image (crowd=1) or no (crowd=0).
- Correlation with proportion topic “Crowd”: 0.58

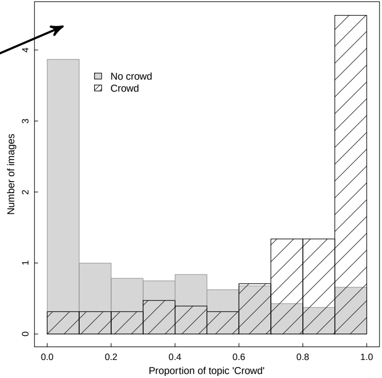
# VALIDATION: HIGH CORRELATION BETWEEN TOPICS AND MANUAL CODING

- Hand-coded sample: presence of medium/big crowd in the image (crowd=1) or no (crowd=0).
- Correlation with proportion topic “Crowd”: 0.58



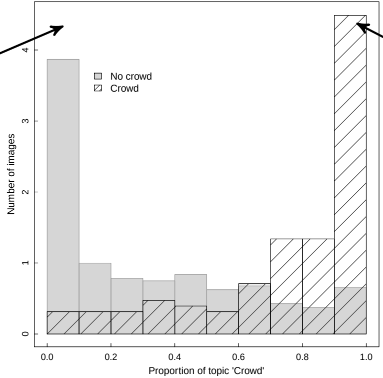
# VALIDATION: HIGH CORRELATION BETWEEN TOPICS AND MANUAL CODING

- Hand-coded sample: presence of medium/big crowd in the image (crowd=1) or no (crowd=0).
- Correlation with proportion topic “Crowd”: 0.58

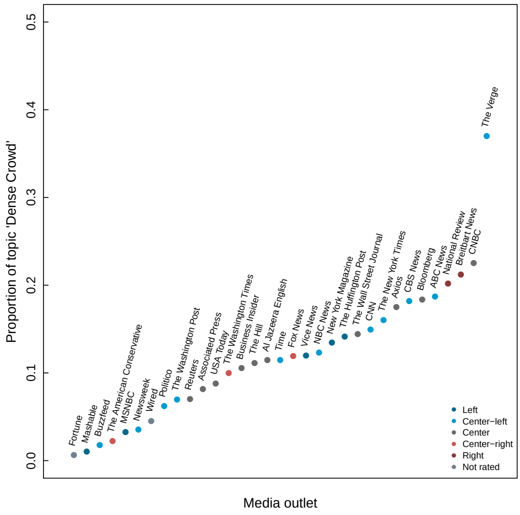



# Validation: high correlation between topics and manual coding

- Hand-coded sample: presence of medium/big crowd in the image (crowd=1) or no (crowd=0).
- Correlation with proportion topic “Crowd”: 0.58


# TOPIC "CROWD" BY MEDIA OUTLET

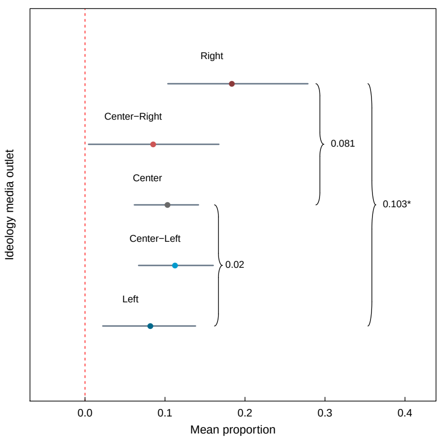


## Factors behind the generation of visual frames

- Estimate the effect of media ideology on prevalence of topic “Crowd”

# FACTORS BEHIND THE GENERATION OF VISUAL FRAMES

- Estimate the effect of media ideology on prevalence of topic "Crowd"



## THINGS TO CONSIDER

- Coherence and symbolism of visual words

## THINGS TO CONSIDER

- Coherence and symbolism of visual words
    - Some tokens are similar in terms of features but not concept

# THINGS TO CONSIDER

- Coherence and symbolism of visual words
  - Some tokens are similar in terms of features but not concept
  - Others are more synonyms than exclusive words

# THINGS TO CONSIDER

- Coherence and symbolism of visual words
  - Some tokens are similar in terms of features but not concept
  - Others are more synonyms than exclusive words
  - Some visual words are plainly “bad”

# THINGS TO CONSIDER

- Coherence and symbolism of visual words
  - Some tokens are similar in terms of features but not concept
  - Others are more synonyms than exclusive words
  - Some visual words are plainly “bad”
- Dimension reduction

# Things to consider

- Coherence and symbolism of visual words
  - Some tokens are similar in terms of features but not concept
  - Others are more synonyms than exclusive words
  - Some visual words are plainly “bad”
- Dimension reduction
  - Mapping of latent concepts to low-dimensional features

## THINGS TO CONSIDER

- Coherence and symbolism of visual words
  - Some tokens are similar in terms of features but not concept
  - Others are more synonyms than exclusive words
  - Some visual words are plainly “bad”
- Dimension reduction
  - Mapping of latent concepts to low-dimensional features
  - Sensitivity of results to feature definition/extraction

# THINGS TO CONSIDER

- Coherence and symbolism of visual words
  - Some tokens are similar in terms of features but not concept
  - Others are more synonyms than exclusive words
  - Some visual words are plainly “bad”
- Dimension reduction
  - Mapping of latent concepts to low-dimensional features
  - Sensitivity of results to feature definition/extraction
- Interpretation of results

# THINGS TO CONSIDER

- Coherence and symbolism of visual words
  - Some tokens are similar in terms of features but not concept
  - Others are more synonyms than exclusive words
  - Some visual words are plainly “bad”
- Dimension reduction
  - Mapping of latent concepts to low-dimensional features
  - Sensitivity of results to feature definition/extraction
- Interpretation of results
  - “tea leaves reading”

# THINGS TO CONSIDER

- Coherence and symbolism of visual words
  - Some tokens are similar in terms of features but not concept
  - Others are more synonyms than exclusive words
  - Some visual words are plainly “bad”
- Dimension reduction
  - Mapping of latent concepts to low-dimensional features
  - Sensitivity of results to feature definition/extraction
- Interpretation of results
  - “tea leaves reading”
- Resources

# THINGS TO CONSIDER

- Coherence and symbolism of visual words
  - Some tokens are similar in terms of features but not concept
  - Others are more synonyms than exclusive words
  - Some visual words are plainly “bad”
- Dimension reduction
  - Mapping of latent concepts to low-dimensional features
  - Sensitivity of results to feature definition/extraction
- Interpretation of results
  - “tea leaves reading”
- Resources
  - Learning curve

# Things to consider

- Coherence and symbolism of visual words
  - Some tokens are similar in terms of features but not concept
  - Others are more synonyms than exclusive words
  - Some visual words are plainly “bad”
- Dimension reduction
  - Mapping of latent concepts to low-dimensional features
  - Sensitivity of results to feature definition/extraction
- Interpretation of results
  - “tea leaves reading”
- Resources
  - Learning curve
  - Time?

# LET'S CODE!

What else is out there?

# Wrapping-up

# A POOL OF OPTIONS

- Create high quality training data and use transfer learning
    - AWS machines, HPC or GPUs [computational power needed]
    - Pre-trained architectures in Google, Amazon, etc.
    - Creating training data: `imglab`
- Pre-canned image detection with API access
    - GoogleVision: `https://cloud.google.com/vision/`, Amazon, Microsoft
        - Labels found in each picture
        - Face detection
        - Emotions
        - Sensitive content (e.g. violence, nudity, etc.)
        - Object detection

# Object detection: Covers of Newspapers

Full set of images


Only Women's March images

# Face detection and emotional content

Good results with little effort...



# FACE DETECTION AND EMOTIONAL CONTENT

Good results with little effort...



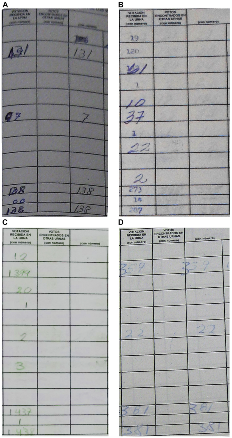

...but also tons of errors(*)


# APPLICATIONS OF COMPUTER VISION TOOLS

# APPLICATIONS OF COMPUTER VISION TOOLS

- Detection of fraud using CNNs

  (Cantú 2019)



# Applications of Computer Vision Tools

- Detection of fraud using CNNs (Cantú 2019)
- Identifying the share of women in political ads and the gender composition to the audiences to which these are deployed (Erfort 2023)

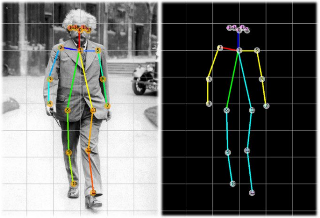
# Applications of Computer Vision Tools

- Detection of fraud using CNNs (Cantú 2019)
- Identifying the share of women in political ads and the gender composition to the audiences to which these are deployed (Erfort 2023)
- Identification of (minor) protests using CNNs and tweets (Steinert-Threlkeld, Chan, and Joo 2022)

Barcelona .654

# Applications of Computer Vision Tools

- Detection of fraud using CNNs (Cantú 2019)
- Identifying the share of women in political ads and the gender composition to the audiences to which these are deployed (Erfort 2023)
- Identification of (minor) protests using CNNs and tweets (Steinert-Threlkeld, Chan, and Joo 2022)
- Nature and reactions/attention to female and male politicians' body language using key points and vocal pitch (Rittman et al. 2023)



## Conclusion

- Images are powerful (and abundant!) elements of frames.

# Conclusion

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication

# CONCLUSION

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication
- The learning curve is steep but it pays off

# CONCLUSION

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication
- The learning curve is steep but it pays off
- Plus, there are many accessible options tailored for many needs

# CONCLUSION

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication
- The learning curve is steep but it pays off
- Plus, there are many accessible options tailored for many needs
- HOWEVER...

# Conclusion

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication
- The learning curve is steep but it pays off
- Plus, there are many accessible options tailored for many needs
- HOWEVER...
- Be thoughtful! Ground your empirics in substantive questions and theoretical insights.

# Conclusion

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication
- The learning curve is steep but it pays off
- Plus, there are many accessible options tailored for many needs
- HOWEVER...
- Be thoughtful! Ground your empirics in substantive questions and theoretical insights.
- Do not use these tools lightly just because you *can*

# CONCLUSION

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication
- The learning curve is steep but it pays off
- Plus, there are many accessible options tailored for many needs
- HOWEVER...
- Be thoughtful! Ground your empirics in substantive questions and theoretical insights.
- Do not use these tools lightly just because you *can*
- Think carefully about what you want the computer to see.

# CONCLUSION

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication
- The learning curve is steep but it pays off
- Plus, there are many accessible options tailored for many needs
- HOWEVER...
- Be thoughtful! Ground your empirics in substantive questions and theoretical insights.
- Do not use these tools lightly just because you *can*
- Think carefully about what you want the computer to see.
    - How are you “teaching” it?

# Conclusion

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication
- The learning curve is steep but it pays off
- Plus, there are many accessible options tailored for many needs
- HOWEVER...
- Be thoughtful! Ground your empirics in substantive questions and theoretical insights.
- Do not use these tools lightly just because you *can*
- Think carefully about what you want the computer to see.
    - How are you “teaching” it?
    - How successful are you? What are the mistakes, successes?

# CONCLUSION

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication
- The learning curve is steep but it pays off
- Plus, there are many accessible options tailored for many needs
- HOWEVER...
- Be thoughtful! Ground your empirics in substantive questions and theoretical insights.
- Do not use these tools lightly just because you *can*
- Think carefully about what you want the computer to see.
  - How are you “teaching” it?
  - How successful are you? What are the mistakes, successes?
  - What about bias?

# CONCLUSION

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication
- The learning curve is steep but it pays off
- Plus, there are many accessible options tailored for many needs
- HOWEVER...
- Be thoughtful! Ground your empirics in substantive questions and theoretical insights.
- Do not use these tools lightly just because you *can*
- Think carefully about what you want the computer to see.
  - How are you “teaching” it?
  - How successful are you? What are the mistakes, successes?
  - What about bias?
  - How complex is your object of interest?

# Conclusion

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication
- The learning curve is steep but it pays off
- Plus, there are many accessible options tailored for many needs
- HOWEVER...
- Be thoughtful! Ground your empirics in substantive questions and theoretical insights.
- Do not use these tools lightly just because you *can*
- Think carefully about what you want the computer to see.
  - How are you "teaching" it?
  - How successful are you? What are the mistakes, successes?
  - What about bias?
  - How complex is your object of interest?
- VALIDATE, VALIDATE, VALIDATE!

# CONCLUSION

- Images are powerful (and abundant!) elements of frames.
- The measurement and analysis of visual components are crucial to have a better understanding of political communication
- The learning curve is steep but it pays off
- Plus, there are many accessible options tailored for many needs
- HOWEVER...
- Be thoughtful! Ground your empirics in substantive questions and theoretical insights.
- Do not use these tools lightly just because you *can*
- Think carefully about what you want the computer to see.
  - How are you “teaching” it?
  - How successful are you? What are the mistakes, successes?
  - What about bias?
  - How complex is your object of interest?
- VALIDATE, VALIDATE, VALIDATE!
- Keep learning and let the creativity take you to infinity and beyond!

# Appendix

# L*A*B COLOR SPACE

- L* = lightness
- a* = chromaticity coordinate (red axis)
- b* = chromaticity coordinate (blue axis)