Workshop on Automated Image Analysis

Andreu Casas
Vrije Universiteit Amsterdam

Organized by the Computational Methods Working Group (CMWG) at the University of Zurich & ETH Zurich

November 5th, 2021

Computational Methods Working Group (CMWG)

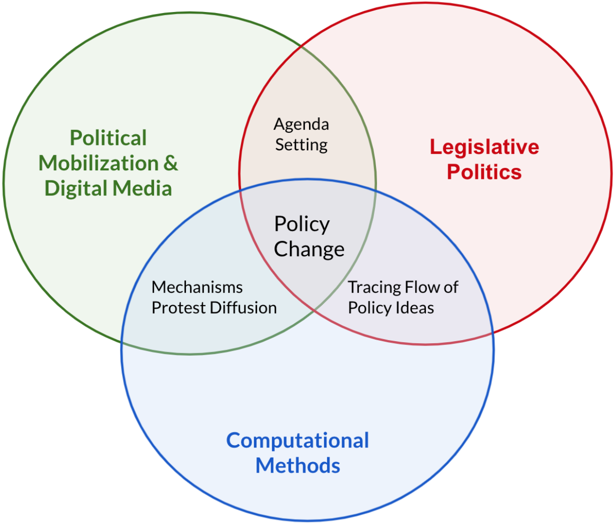
Program

9:30-10:00	Welcome, introductions and housekeeping
10:00-10:30	Introduction to Images as Data in the Social Sciences
10:30-10:35	(*5-min. Break*)
10:35-11:20	Introduction to Neural Nets and Computer Vision
11:20-11:30	(*10-min Break*)
11:30-12:15	Hands-on Module #1: Image processing
12:15-13:00	(*45-min. Lunch Break*)
13:00-13:45	Hands-on Module #2: Image classification
13:45-14:00	(*15-min.* Break)
14:00-14:45	Hands-on Module #3: Face detection/recognition
14:45-end	Discussion and Project consultation

Program

9:30-10:00	**Welcome, introductions and housekeeping**
10:00-10:30	Introduction to Images as Data in the Social Sciences
10:30-10:35	(*5-min. Break*)
10:35-11:20	Introduction to Neural Nets and Computer Vision
11:20-11:30	(*10-min Break*)
11:30-12:15	Hands-on Module #1: Image processing
12:15-13:00	(*45-min. Lunch Break*)
13:00-13:45	Hands-on Module #2: Image classification
13:45-14:00	(*15-min.* Break)
14:00-14:45	Hands-on Module #3: Face detection/recognition
14:45-end	Discussion and Project consultation

Andreu Casas



a.casassalleras@vu.nl | @CasAndreu | http://andreucasas.com/

Who are you?

- Name
- Institution
- Research interest
- Why are you interested in Images as Data?

Housekeeping

- Dense program: please feel free to ask questions, clarification, for a break, etc., at any time.
- I'm doing some new things: hopefully timing won't be off by too much.
- *Google Colab* for the Hands-on Modules.
  - You need to have a google account.
  - You must have received (via email) a zip file with the data/code for the tutorials.
  - Unzip the file, and upload the folder in there into your Google Drive (within the main "`My Drive`" folder).
- This workshop is for beginners. I assume...
  - ... people are familiar with `python` programming.
  - ... people have little-to-no knowledge of computer vision and deep learning.

Program

9:30-10:00	~~**Welcome, introductions and housekeeping**~~
10:00-10:30	**Intro to Images as Data in the Social Sciences**
10:30-10:35	(*5-min. Break*)
10:35-11:20	Introduction to Neural Nets and Computer Vision
11:20-11:30	(*10-min Break*)
11:30-12:15	Hands-on Module #1: Image processing
12:15-13:00	(*45-min. Lunch Break*)
13:00-13:45	Hands-on Module #2: Image classification
13:45-14:00	(*15-min.* Break)
14:00-14:45	Hands-on Module #3: Face detection/recognition
14:45-end	Discussion and Project consultation

Images as Data in the Social Sciences

Outline

1. Why do Images Matter?
2. Types of Existing Research with Images as Data
3. Available Automated Image Analysis Methods
   - what we'll cover
   - what we'll **not** cover
4. Good practices and limitations

Why do Images Matter?

People are more likely to pay attention to visuals

IMMIGRATION

How America Got to 'Zero Tolerance' on Immigration

Battles have raged within the White House over family separations, ICE raids and President Trump's obsession with a wall.

Together, they have remade homeland security.

15m ago 393 comments

Mr. Trump's approach follows a model from Europe and Australia, our Interpreter columnists write.

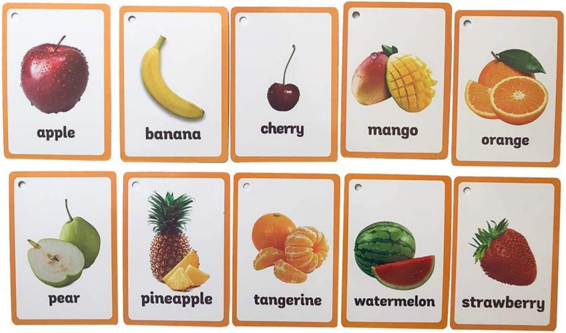
3h ago

Kirsten Luce

Dahmen (2012) "*Photographic Framing in the Stem Cell Debate*"

Why do Images Matter?

People are more likely to recall information learned through visuals



Paivio et al. (1968) "*Why are pictures easier to recall than words?*"

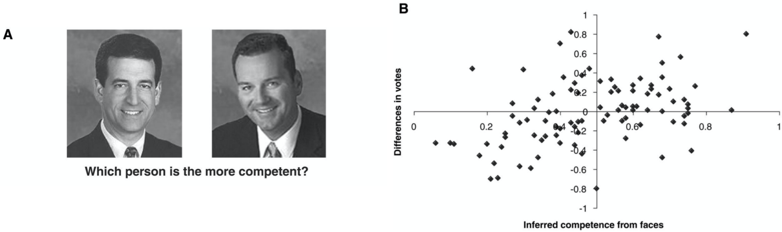
Why do Images Matter?
Visuals evoke stronger emotional reactions



Grabe Bucy (2009) "*Images Bite Politics*"

Why do Images Matter?

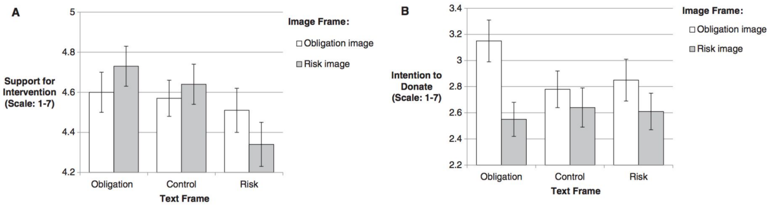
Image effects in **politics**: images $\rightarrow$ inference of competence $\rightarrow$ voting



Todorov et al. (2009) "*Inferences of Competence from Faces Predict Election Outcomes*"

Why do Images Matter?

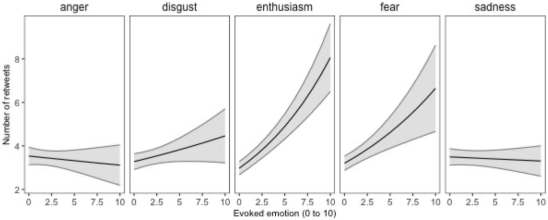
Image effects in **politics**: images → framing → attitudes



Powell et al. (2015) "A Clearer Picture"

Why do Images Matter?

Image effects in **politics**: images $\rightarrow$ emotions $\rightarrow$ **mobilization**



Casas & Webb Williams (201) "Images That Matter"

Why do Images Matter?

Images are more central than even in our life

5,987,390,092 Google searches today	**5,798,222** Blog posts written today	**601,863,479** Tweets sent today
5,719,819,412 Videos viewed today on YouTube	**68,873,768** Photos uploaded today on Instagram	**122,683,021** Tumblr posts today
2,967,305,806 Facebook active users	**1,032,597,179** Google+ active users	**380,514,482** Twitter active users

`https://www.internetlivestats.com/`

Types of Existing Research with Images as Data
Causal Framework

- Images as independent variable
  - Casas and Webb Williams (PRQ 2018): *Which Black Lives Matter images mobilized more supporters?*
- Images as dependent variable
  - Michelle Torres (working paper): *How do different news organizations choose different pictures to accompany articles about Black Lives Matter?*

Types of Existing Research with Images as Data
As a Measurement Strategy

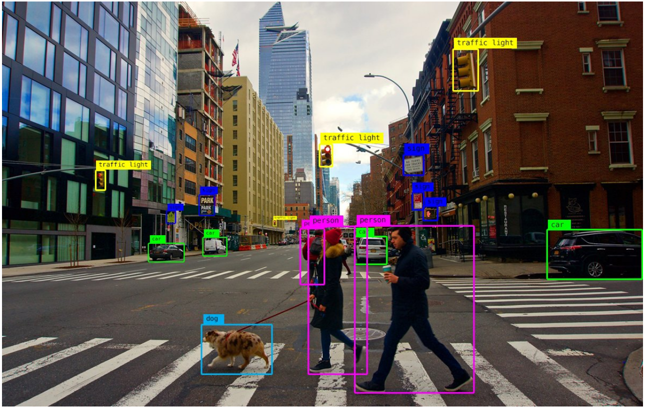
- Images can contain information about electoral incidents and fraud (Callen and Long (2015); Cantú (working paper); Mebane et al (working paper))
- Images can help us identify and classify protest events (Zhang and Pan (2018), Won, Steinert-Threlkeld and Joo (2017))
- Nighttime lights imagery as a proxy for economic development (many authors)
- Digitized historical maps as evidence of road quality variation (Hunziker et al (working paper))
- Videos/Images can help us measure cooperation in legislative politics (Dietrich 2020)

Types of Existing Research with Images as Data
Methodological contributions

- Unsupervised clustering (Casas et al.(working paper); and others)
- Limitations & biases (Schwemmer et al. 2020)
- Methodological reviews (Webb Williams et al. 2020; Torres & Cantu 2021)

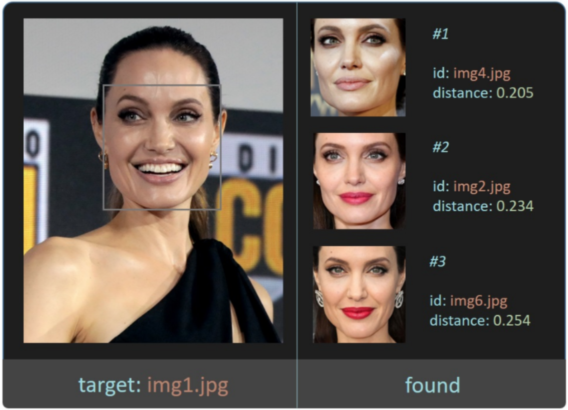
Available Automated Image Analysis Methods

Object detection & recognition



Available Automated Image Analysis Methods

Face detection & recognition



Available Automated Image Analysis Methods

Face analysis

Available Automated Image Analysis Methods

Image Similarity

Available Automated Image Analysis Methods

Unsupervised Clustering

(1)
(2)
(3)
(4)
(5)
(6)
(7)
(8)
(9)
(10)

Available Automated Image Analysis Methods
And many others...

- Text extraction (OCR)
- Caption generation
- Sentiment analysis (evoked emotions)
- Visual aesthetics analysis
- etc...

Available Automated Image Analysis Methods

Today in the hands-on modules we'll mainly focus on...

- Some basic image (& video) manipulation/processing
- Image classification
- Face detection/recognition/analysis

Pitfalls and limitations

Important Warnings

- Limitations of commercial (and off-the-shelf) services
- Biases in AI
- Data privacy
- General ethics

Program

9:30-10:00	~~**Welcome, introductions and housekeeping**~~
10:00-10:30	~~**Intro to Images as Data in the Social Sciences**~~
10:30-10:35	**(5-min. Break)**
10:35-11:20	Introduction to Neural Nets and Computer Vision
11:20-11:30	(*10-min Break*)
11:30-12:15	Hands-on Module #1: Image processing
12:15-13:00	(*45-min. Lunch Break*)
13:00-13:45	Hands-on Module #2: Image classification
13:45-14:00	(*15-min.* Break)
14:00-14:45	Hands-on Module #3: Face detection/recognition
14:45-end	Discussion and Project consultation

Program

9:30-10:00	~~**Welcome, introductions and housekeeping**~~
10:00-10:30	~~**Intro to Images as Data in the Social Sciences**~~
10:30-10:35	~~**(5-min. Break)**~~
10:35-11:20	**Introduction to Neural Nets and Computer Vision**
11:20-11:30	(*10-min Break*)
11:30-12:15	Hands-on Module #1: Image processing
12:15-13:00	(*45-min. Lunch Break*)
13:00-13:45	Hands-on Module #2: Image classification
13:45-14:00	(*15-min.* Break)
14:00-14:45	Hands-on Module #3: Face detection/recognition
14:45-end	Discussion and Project consultation

Intro to Neural Nets and Computer Vision

Outline

1 Neural Networks
- Artificial Intelligence. *Sounds fancy, but how does it work?*

2 Computer Vision
- Convolutional Neural Networks. *The Basics.*

Neural Networks
Why?

In the last few years Artificial Neural Networks and deep learning have drastically improved machine-learning performance.

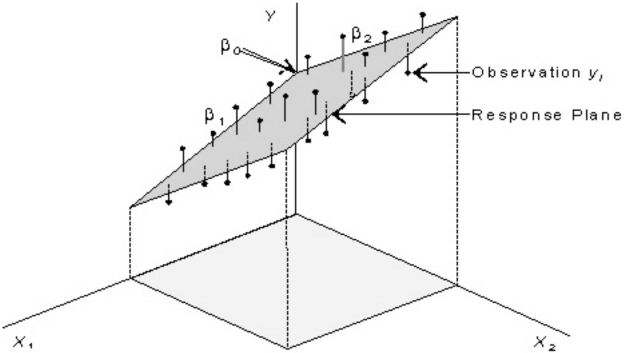
- Speech-recognition (e.g. Siri, Echo, Alexa)
- Translation (e.g. Google translator)
- Image recognition (e.g. Facebook's facial recognition photo tagging)

Neural Networks

In “conventional” machine learning, we only use a single parameter matrix: 1 variable = 1 coefficient.

Linear Model: regression formula

$$y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon_i$$



Neural Networks

In "conventional" machine learning, we only use a single parameter matrix: 1 variable = 1 coefficient.

Linear Model: compact matrix form

$$\mathbf{Y} = \mathbf{X}\beta$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} \\ 1 & x_{21} & x_{22} \\ 1 & x_{31} & x_{22} \\ 1 & x_{41} & x_{22} \\ \vdots & \vdots & \\ 1 & x_{n1} & x_{n2} \end{bmatrix} * \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix}$$

Neural Networks

In "conventional" machine learning, we only use a single parameter matrix: 1 variable = 1 coefficient.

- Interested in finding the parameter matrix β that minimizes predictive error
- This is easy when using a Least Square regression because there is an analytic solution

$$\beta = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'y$$

- We can use MLE to find this parameter matrix for more complex general linear models

$$\mathbf{Y} = logit(\mathbf{X}\beta)$$

Neural Networks

Conventional machine learning only take us so far... what about extending the learning process?

What if we use the output of a first model as input of a second model...

$$\hat{\mathbf{Y}}_{\mathbf{2}} = \mathbf{X}\beta_1$$

$$\hat{\mathbf{Y}} = \mathbf{X}_2\beta_2$$

where

$$\hat{\mathbf{Y}}_{\mathbf{2}} = \mathbf{X}_{\mathbf{2}}$$

and we try to minimize $\mathbf{Y} - \hat{\mathbf{Y}}$ instead of $\mathbf{Y} - \hat{\mathbf{Y}}_{\mathbf{2}}$

This is what we call a Neural Network or Artificial Neural Network!

Neural Networks

Matrix multiplication is the key to understand neural nets!

Remember these 2 key principles of matrix multiplication:

1. the number of columns in the first matrix has to be the same than the number of rows in the second matrix
2. the number of rows of the resulting matrix will equal the number of rows of the first matrix, and the number of columns will equal the number of columns of the second matrix

$$A[n, k] * B[k, z] = C[n, z]$$

Neural Networks

Matrix multiplication is the key to understand neural nets!

Instead of a simple linear or general linear model we can have a model that looks like this...

$$\text{Sigmoid}(\mathbf{X}[1000, 4]\ \beta_1[4, 250])\ \beta_2[250, 1] = \mathbf{Y}[1000, 1]$$

...

(1) $\mathbf{X}[1000, 4]\ \beta_1[4, 250] \rightarrow \mathbf{X_2}[1000, 250]$

(2) $\text{Sigmoid}(\mathbf{X_2}[1000, 250]) \rightarrow \mathbf{X_{2b}}[1000, 250]$

(3) $\mathbf{X_{2b}}[1000, 250]\ \beta_2[250, 1] \rightarrow \hat{\mathbf{Y}}[1000, 1]$

We calculate the parameters in the matrices β_1 and β_2 using e.g. Stochastic Gradient Descent $\rightarrow$ iterating until convergence

Neural Networks

Some basic terminology... different words for some familiar concepts

- **input layer**: the original data matrix ($\mathbf{X}$)
- **weight/s**: a single parameter (β_{ij}) / parameter matrix (β)
- **bias**: the intercept parameter matrix (α or β_0)
- **ReLu, Sigmoid, Tanh**: non-linear transformation we apply to $\mathbf{X}$ matrices. Also known as **activation functions**
- **hidden layer**: $\mathbf{X_2}, \mathbf{X_3}, ...$ a new intermediate representation of the input
- **loss function**: the function we want to minimize (e.g. $\mathbf{\hat{Y}} - \mathbf{Y}$)
- **regularization**: transformations we apply to the loss function (e.g. $|\mathbf{\hat{Y}} - \mathbf{Y}| \rightarrow$ L1 and $(\mathbf{\hat{Y}} - \mathbf{Y})^2 \rightarrow$ L2) or the variables/columns of the input matrix
- **dropout**: setting some β_{ij} from a β matrix to 0 at random
- **forward propagation**: performing all matrix multiplications
- **backpropagation**: calculating Stochastic Gradient Descent

Neural Networks

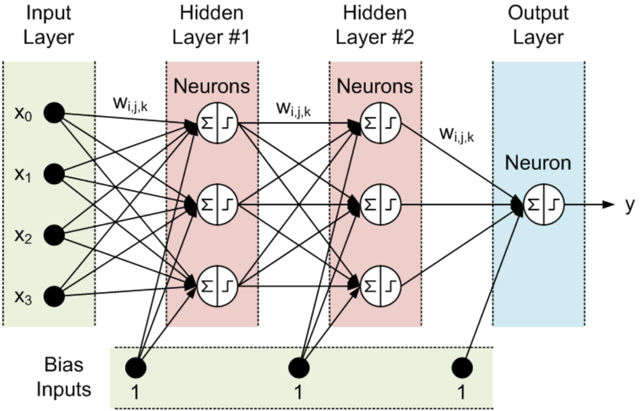
Some more terminology and... hyperparemeters, the dark mysteries of neural nets

- **graph**: a model
- **train-test-validation split**: 80-10-10? 50-25-25?
- **batch size**: the number of training observations we use for training in a given iteration
- **epochs**: number of training iterations
- **dropout rate**: the probability of re-initializing a given weight
- **learning rate**: by how much we update the weights at each training iteration

There are some conventions people follow. Since we are preforming supervised training, we always look for the hyperparemeters that achieve the highest out-of-sample accuracy.

Neural Networks

Neural nets are often represented this way



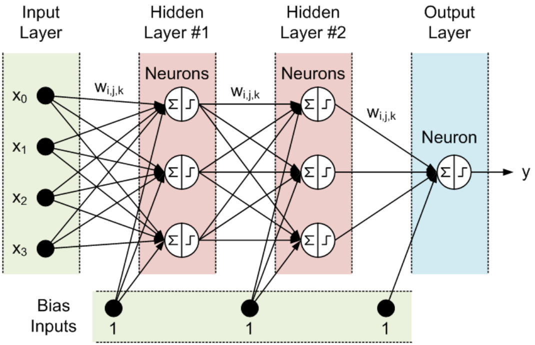
To be fair, the term “deep learning” should be used only when the neural networks have a several hidden layers. But how deep does a neural net need to be in order to be considered deep learning?

Neural Networks

Fine tuning or transfer learning

Slightly tweaking an already trained neural net to predict a different outcome

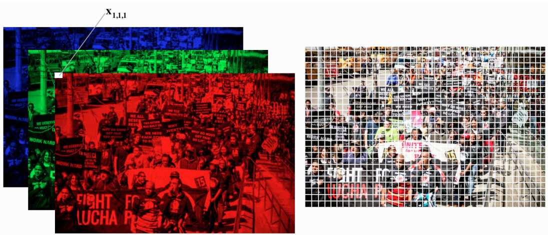
- Retraining the whole neural net with new data
- Retraining part of the neural net with new data
- Adding or changing layers



Convolutional Neural Networks. *The Basics.*

Convolutional Neural Nets for Computer Vision
Two main differences

(1) Images as inputs: 3-dimensional matrices (width x height x depth)



$\mathbf{X} =$

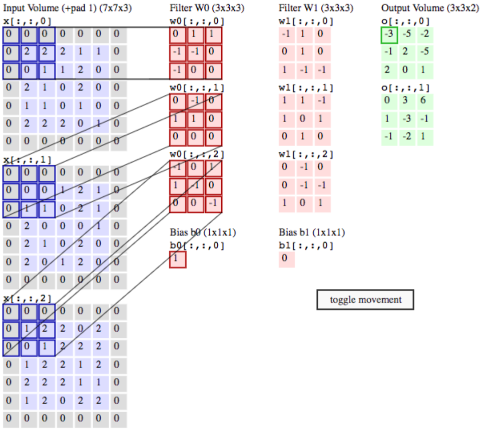
$$\begin{bmatrix} x_{111} & x_{112} & \dots & x_{11n} \\ x_{121} & x_{122} & \dots & x_{12n} \\ x_{131} & x_{132} & \dots & x_{13n} \\ x_{141} & x_{142} & \dots & x_{14n} \\ \vdots & \vdots & & \\ x_{1n1} & x_{1n2} & \dots & x_{1nn} \end{bmatrix}, \begin{bmatrix} x_{211} & x_{212} & \dots & x_{21n} \\ x_{221} & x_{222} & \dots & x_{22n} \\ x_{231} & x_{232} & \dots & x_{23n} \\ x_{241} & x_{242} & \dots & x_{24n} \\ \vdots & \vdots & & \\ x_{2n1} & x_{2n2} & \dots & x_{2nn} \end{bmatrix}, \begin{bmatrix} x_{311} & x_{312} & \dots & x_{31n} \\ x_{321} & x_{322} & \dots & x_{32n} \\ x_{331} & x_{332} & \dots & x_{33n} \\ x_{341} & x_{342} & \dots & x_{34n} \\ \vdots & \vdots & & \\ x_{3n1} & x_{3n2} & \dots & x_{3nn} \end{bmatrix}$$

Convolutional Neural Nets for Computer Vision

Two main differences

(2) Convolutional layers: weights (**filters**) are not connected to the whole **input volume**: convolution.

Click here for a full visualization by the Stanford cs231 folks.



Convolutional Neural Nets for Computer Vision
Some new terminology... and more hyperparameters

- **input volume**: a 3-dimensional input
- **convolutional layer**: a 4-dimensional parameter layer where convolutional filters are applied to the input volume; of size FxFxNxK where F is the width and height of the filter, N is the number of filter dimensions, and K is the number of filters $\rightarrow$ 3x3x3x2 in the previous example
- **stride**: the number of pixels we move the filter at a time. This is 2 in the previous example
- **zero-padding**: adding zeros around the input border (often done to avoid deforming input images)
- **pooling layer**: a layer where we reduce the size the output of a convolutional layer. From 224x224x3x64 to 112x112x3x64 for example.

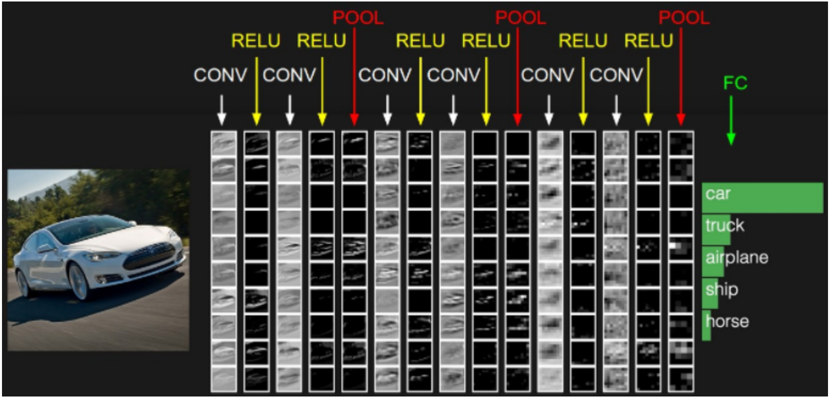
Convolutional Neural Nets for Computer Vision

Some new terminology... and more hyperparameters

- **fully connected layer**: a layer of weights that is connected to the whole input volume. These are usually at the end of a network.
- **softmax**: a multi-class classifier. This is basically a multinomial logit model that uses the output of the last fully-connected layer to predict the final classes of interest

Convolutional Neural Nets for Computer Vision

This is how a ConvNet looks like



Convolutional Neural Nets for Computer Vision
VGG16's architecture

```
INPUT: [224x224x3]        memory:  224*224*3=150K   weights: 0
CONV3-64: [224x224x64]  memory:  224*224*64=3.2M   weights: (3*3*3)*64 = 1,728
CONV3-64: [224x224x64]  memory:  224*224*64=3.2M   weights: (3*3*64)*64 = 36,864
POOL2: [112x112x64]  memory:  112*112*64=800K   weights: 0
CONV3-128: [112x112x128]  memory:  112*112*128=1.6M   weights: (3*3*64)*128 = 73,728
CONV3-128: [112x112x128]  memory:  112*112*128=1.6M   weights: (3*3*128)*128 = 147,456
POOL2: [56x56x128]  memory:  56*56*128=400K   weights: 0
CONV3-256: [56x56x256]  memory:  56*56*256=800K   weights: (3*3*128)*256 = 294,912
CONV3-256: [56x56x256]  memory:  56*56*256=800K   weights: (3*3*256)*256 = 589,824
CONV3-256: [56x56x256]  memory:  56*56*256=800K   weights: (3*3*256)*256 = 589,824
POOL2: [28x28x256]  memory:  28*28*256=200K   weights: 0
CONV3-512: [28x28x512]  memory:  28*28*512=400K   weights: (3*3*256)*512 = 1,179,648
CONV3-512: [28x28x512]  memory:  28*28*512=400K   weights: (3*3*512)*512 = 2,359,296
CONV3-512: [28x28x512]  memory:  28*28*512=400K   weights: (3*3*512)*512 = 2,359,296
POOL2: [14x14x512]  memory:  14*14*512=100K   weights: 0
CONV3-512: [14x14x512]  memory:  14*14*512=100K   weights: (3*3*512)*512 = 2,359,296
CONV3-512: [14x14x512]  memory:  14*14*512=100K   weights: (3*3*512)*512 = 2,359,296
CONV3-512: [14x14x512]  memory:  14*14*512=100K   weights: (3*3*512)*512 = 2,359,296
POOL2: [7x7x512]  memory:  7*7*512=25K  weights: 0
FC: [1x1x4096]  memory:  4096  weights: 7*7*512*4096 = 102,760,448
FC: [1x1x4096]  memory:  4096  weights: 4096*4096 = 16,777,216
FC: [1x1x1000]  memory:  1000 weights: 4096*1000 = 4,096,000

TOTAL memory: 24M * 4 bytes ~= 93MB / image (only forward! ~*2 for bwd)
TOTAL params: 138M parameters
```

Onto the hands-on modules.
Let's practice!

Program

9:30-10:00	~~**Welcome, introductions and housekeeping**~~
10:00-10:30	~~**Intro to Images as Data in the Social Sciences**~~
10:30-10:35	~~**(5-min. Break)**~~
10:35-11:20	~~**Introduction to Neural Nets and Computer Vision**~~
11:20-11:30	**(10-min Break)**
11:30-12:15	Hands-on Module #1: Image processing
12:15-13:00	(*45-min. Lunch Break*)
13:00-13:45	Hands-on Module #2: Image classification
13:45-14:00	(*15-min.* Break)
14:00-14:45	Hands-on Module #3: Face detection/recognition
14:45-end	Discussion and Project consultation