Graph Constructions

Amusing Graphs and How to Build Them

Ng Yen Kaow

Constructing graphs from...

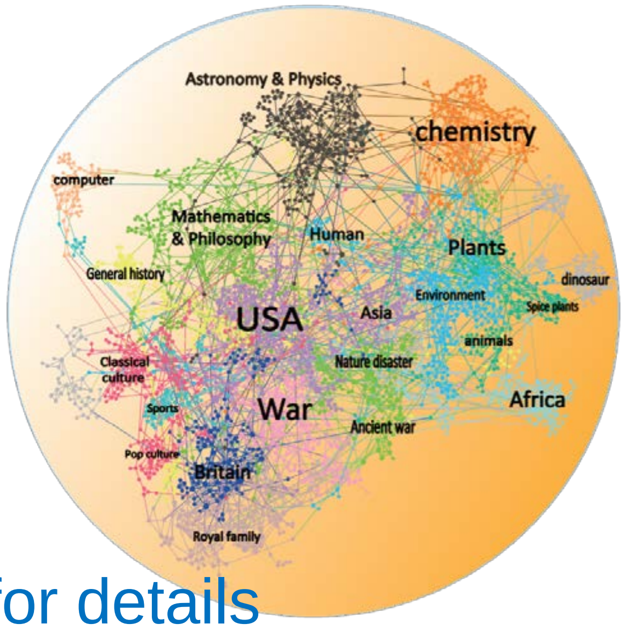
- Natural links
  - Social networks or internet links
- Feature vectors
  - See “Similarity and Dissimilarity.pptx”
- Strings (biosequences)
  - Hamming/Edit distance
- Named objects (or words)
  - Single word or object (semantic similarity)
  - Documents or bag-of-words (Jaccard distance)
- Images / Videos
  - Graph from images using Siamese Network
  - Graph from objects identified in images (CNN)
- Spectrogram representation of sound

Graph from natural links

- Twitter/Facebook
  - Friend/follower relations are edges

- Webpages
  - Hyperlinks are edges

- Wikipedia entries
  - Work by students in class of 2018
  - Each entry is a vertex
  - Hyperlinks are edges
  - See “Files > Projects” in Luminus for details



Graph from feature vectors

- Features are, for example, the stats of an object or a person
  - Two objects that have similar stats can be considered similar

  - More in "Similarity and Dissimilarity.pptx"
- Soccer player stats
  - Work by students in class of 2019
  - Each soccer player is a vertex
  - An edge for two players with similar stats
  - See "Files > Projects" in Luminus for details

Graph from strings/biosequences

- Hamming / edit distance
- Many other measures
  - Global alignment
    - Needleman-Wunsch algorithm
  - Local alignment
    - Smith-Waterman algorithm
  - Blast family of methods
    - For very long sequences blast-based methods have become the standard

Both are straight-forward dynamic programming methods

- No past project(s) by students so far

Graph from named objects

- Online shops

  Data can be downloaded manually (for small project) or automated, e.g. https://www.blog.datahut.co/post/scraping-amazon-reviews-python-scrapy

  - (The goods of) each shop is a vertex
    - Form edge if two shops if they share similar goods
  - (The comments of) each person is a vertex
    - Form edge if two persons wrote very similar comments

- Recipes

  Data can be downloaded manually (for small project) or from databases, e.g. https://esha.com/resources/additional-databases/

  - (The ingredients of) each recipe is a vertex
    - Form edge if two recipes use similar ingredients

- Movies (we will show this example)
  - (The cast of) each movie is a vertex
    - Form edge if two movies have similar casts

Distance between named objects

- Distance between two sets of objects/words
  - Set of objects = bag-of-words or a document
  - Jaccard Index
    - The Jaccard index between two sets of objects A and B

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

    - $J(A,B)$ is metric

- Distance between two words
  - Embedded as a vector
    - Word Embedding https://en.wikipedia.org/wiki/Word_embedding
    - Word2Vec https://en.wikipedia.org/wiki/Word2vec
  - Huge topic in NLP
    - https://en.wikipedia.org/wiki/Semantic_similarity

Named objects example: Movies

- Data from themoviedb.org
  - Open database
  - Convenient Python library

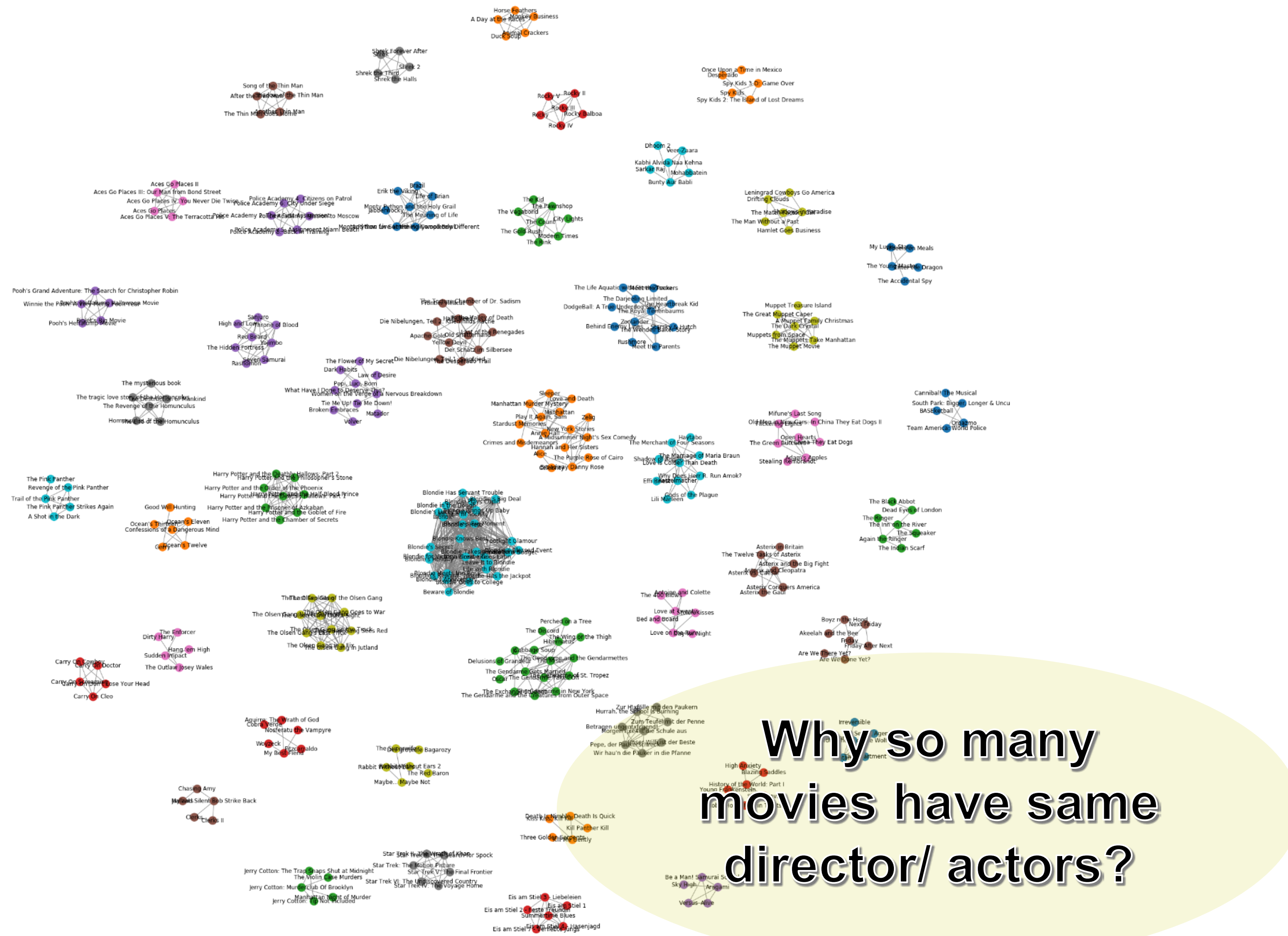
```
import tmdbsimple as tmdb

tmdb.API_KEY = "___API_KEY___" # Use API key given

movie = tmdb.Movies(100) # Get movie of index 100
credits = movie.credits() # Get credits of movie
```

- Data collection done ~2017
  - Retrieval rate: 1 second ~ 3 movies
  - Collected movies from index 1 to 15000
    - 9042 movies retrieved (no data for 5958 indexes)

Named objects example: Movies

- Graph construction
  - Bag-of-words
  - Use a simple measure for edges
    - Place an edge between two movies if they share at least 3 cast members (including director)
    - Seems more natural than Jaccard Index for this situation
  - Resultant graph
    - 1255 vertices
    - 1525 edges
- Community discovery
  - Girvan-Newman clustering

Movie results (using palsgraph)



Why so many movies have same director/ actors?

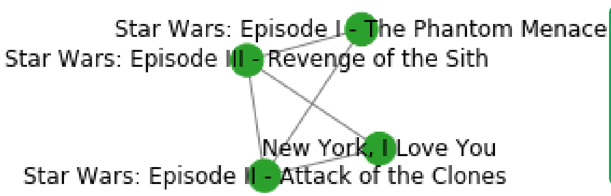
Movie clustering reason: Sequels

- Pirates of the Caribbean: The Curse of the Black Pearl (2003)
- Pirates of the Caribbean: Dead Man's Chest (2006)
- Pirates of the Caribbean: At World's End (2007)

Note: These are movies. Not Disneyland rides

- Back to the Future (1985)
- Back to the Future Part II (1989)
- Back to the Future Part III (1990)

Note: This is old Star Wars.
Yoda looks like this:



Not this

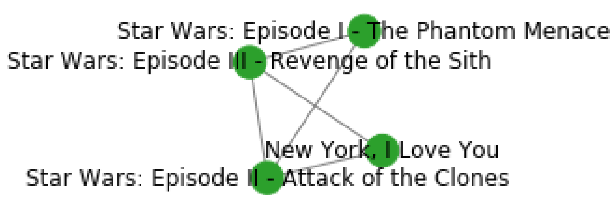
- Star Wars: Episode I - The Phantom Menace (1999)
- Star Wars: Episode II - Attack of the Clones (2002)
- Star Wars: Episode III - Revenge of the Sith (2005)
- New York, I Love You (2008)

Is this a mistake?

Movie clustering reason: Sequels

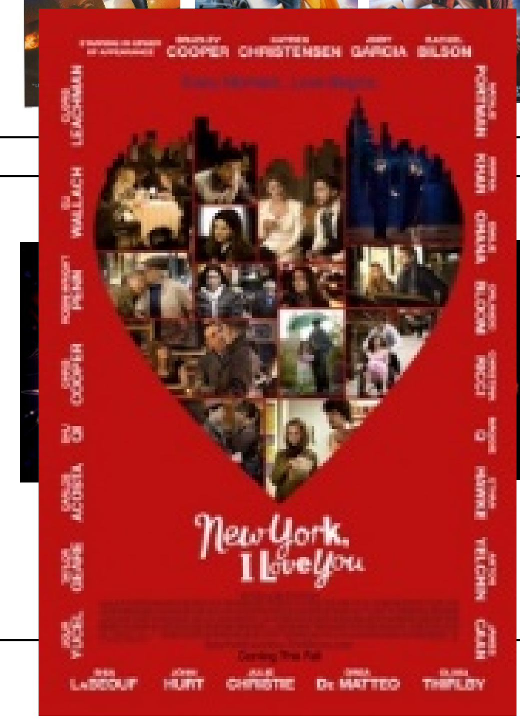
- Pirates of the Caribbean: The Curse of the Black Pearl (2003)
- Pirates of the Caribbean: Dead Man's Chest (2006)
- Pirates of the Caribbean: At World's End (2007)

- Back to the Future (1985)
- Back to the Future Part II (1989)
- Back to the Future Part III (1990)



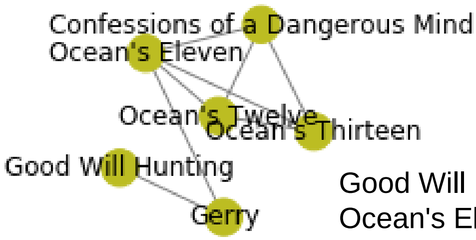
- Star Wars: Episode I - The Phantom Menace (1999)
- Star Wars: Episode II - Attack of the Clones (2002)
- Star Wars: Episode III - Revenge of the Sith (2005)
- New York, I Love You (2008)

Is this a mistake?



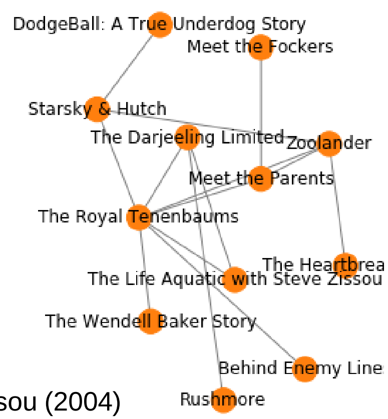
Reason: Actors community

- Matt Damon, George Clooney



Good Will Hunting (1997)
Ocean's Eleven (2001)
Gerry (2002)
Confessions of a Dangerous Mind (2002)
Ocean's Twelve (2004)
Ocean's Thirteen (2007)

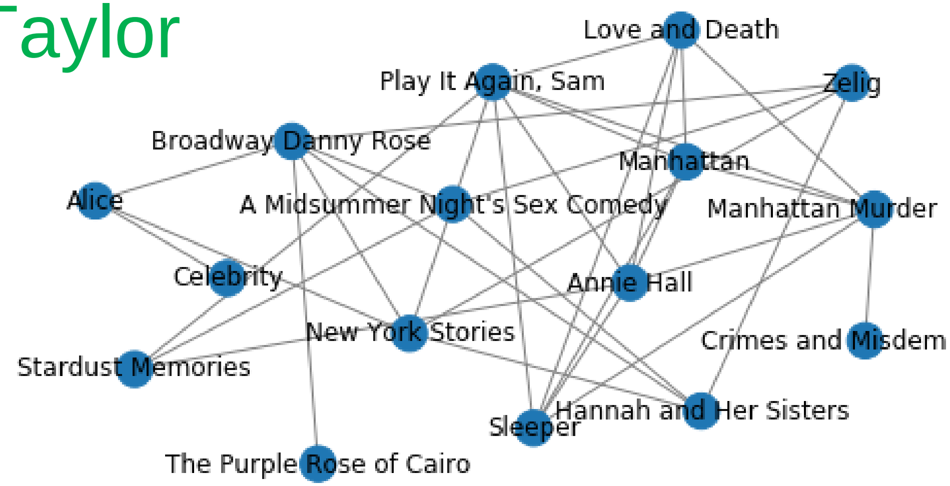
- Owen Wilson, Bill Murray, Ben Stiller



Rushmore (1998)
Meet the Parents (2000)
Behind Enemy Lines (2001)
Zoolander (2001)
The Royal Tenenbaums (2001)
The Life Aquatic with Steve Zissou (2004)
Meet the Fockers (2004)
Starsky & Hutch (2004)
The Wendell Baker Story (2005)
The Darjeeling Limited (2007)
The Heartbreak Kid (2007)

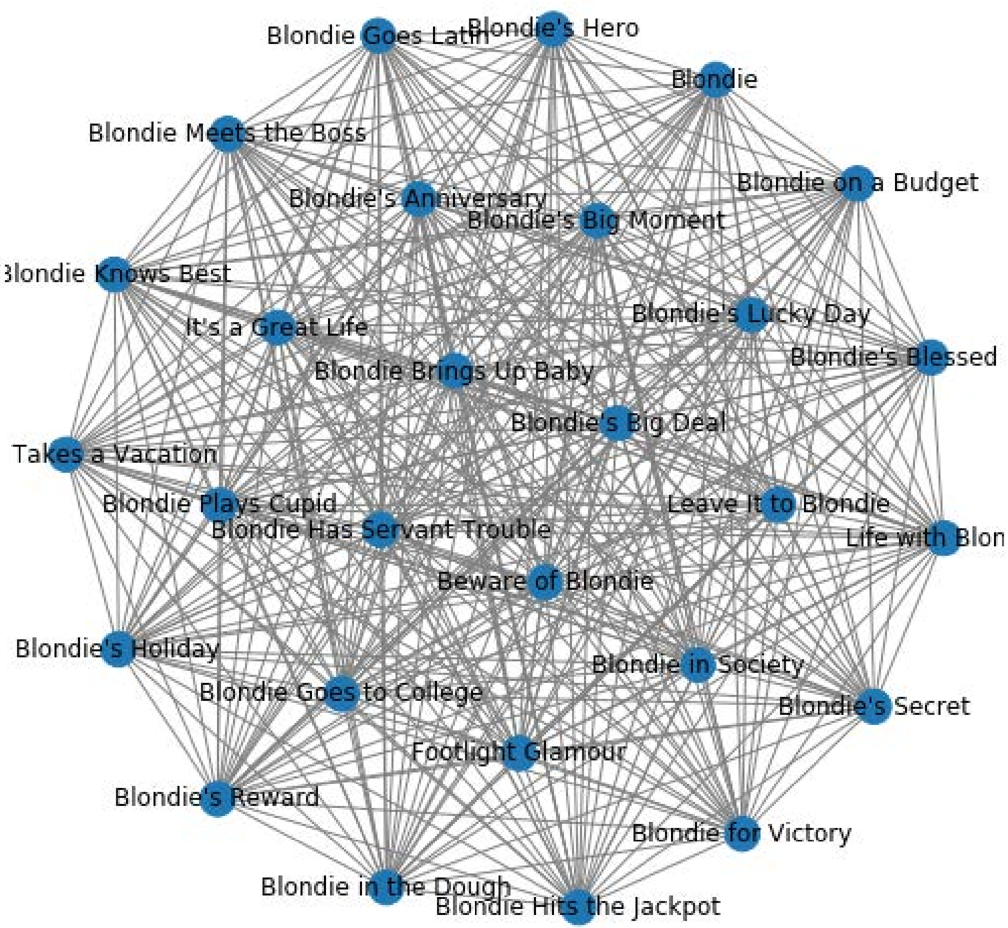
Reason: Director's preference

- Woody Allen, Juliet Taylor



Play It Again, Sam (1972)
Sleeper (1973)
Love and Death (1975)
Annie Hall (1977)
Manhattan (1979)
Stardust Memories (1980)
A Midsummer Night's Sex Comedy (1982)
Zelig (1983)
Broadway Danny Rose (1984)
The Purple Rose of Cairo (1985)
Hannah and Her Sisters (1986)
Crimes and Misdemeanors (1989)
New York Stories (1989)
Alice (1990)
Manhattan Murder Mystery (1993)
Celebrity (1998)

Giant cluster



Blondie (1938)
Blondie Brings Up Baby (1939)
Blondie Takes a Vacation (1939)
Blondie Meets the Boss (1939)
Blondie Has Servant Trouble (1940)
Blondie on a Budget (1940)
Blondie Plays Cupid (1940)
Blondie in Society (1941)
Blondie for Victory (1942)
Blondie Goes Latin (1941)
Blondie Goes to College (1942)
Blondie's Blessed Event (1942)
Footlight Glamour (1943)
It's a Great Life (1943)
Leave It to Blondie (1945)
Blondie Knows Best (1946)
Blondie's Lucky Day (1946)
Life with Blondie (1945)
Blondie in the Dough (1947)
Blondie's Anniversary (1947)
Blondie's Big Moment (1947)
Blondie's Holiday (1947)
Blondie's Reward (1948)
Blondie's Secret (1948)
Blondie Hits the Jackpot (1949)
Blondie's Big Deal (1949)
Beware of Blondie (1950)
Blondie's Hero (1950)

Giant cluster

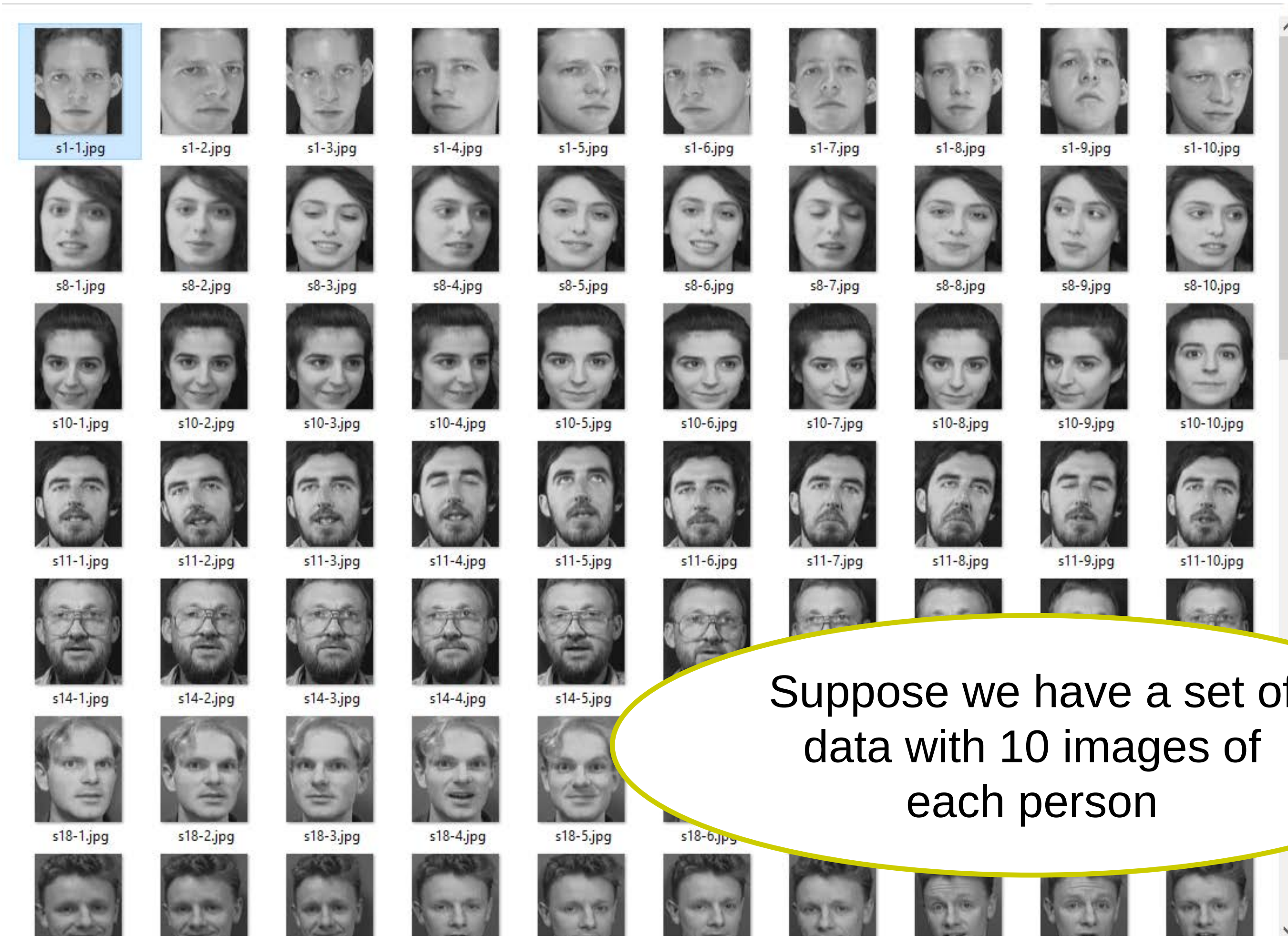
Project(s) in the past

- A few projects in the past were based on named objects
  - News titles (Data: Jinri Toutiao, Sohu, Reuters)
  - Harry Potter characters (Data: Harry Potter books)
  - Depressed posts on Weibo (Data: Weibo)
  - Github accounts (Data: Github)
  - YouTube titles (Data: YouTube)
  - Movie fans’ book preference (Data: Douban)

- See “Files > Projects” in Luminus for details

Graph from Siamese Network

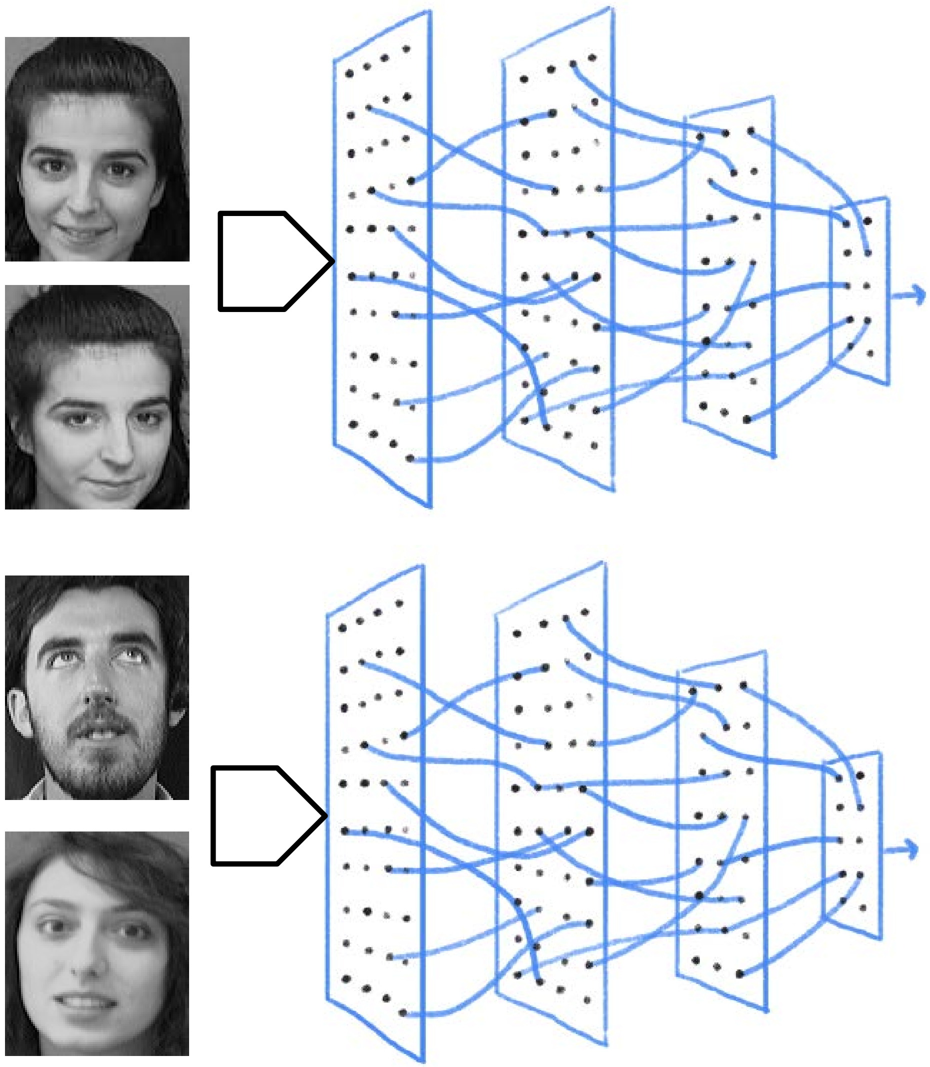
- Build similarity matrix from scores with Siamese network



Suppose we have a set of data with 10 images of each person

Graph from Siamese Network

- We want a Siamese network that does this



0.01 (similar)

1000000 (dissimilar)

We will show you how to train such a network later in colab

Graph from Siamese Network

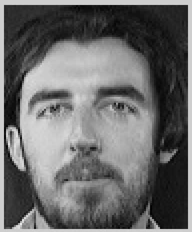
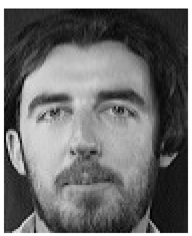
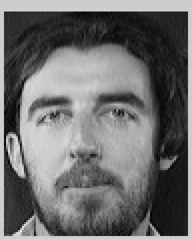
- Using the trained network a distance can be estimated between two classes

For every pair of images each from the two classes compute their distance

Find the average, which indicate how similar or dissimilar the two classes of images are

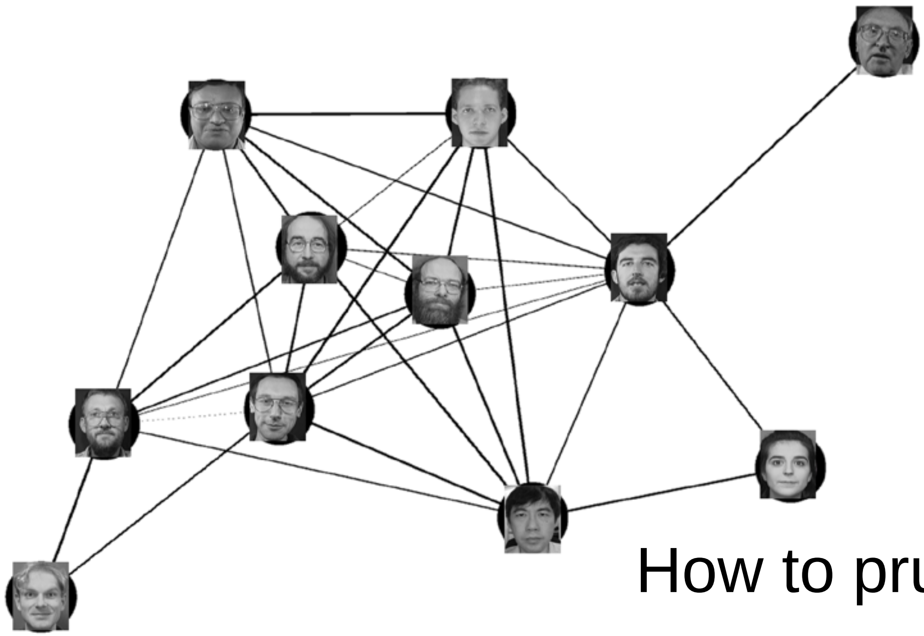
Graph from Siamese Network

- By getting the distances between every 2 classes…

Class 1	Class 2	
		5.53
		1.16
		2.23

Graph from Siamese Network

- …we can construct a network by pruning off edges



How to prune edges

- Method 1: Keep/throw edges
  - Method 1.1: Use a cutoff threshold
  - Method 1.2: Majority voting (next slide)
- Method 2: Use all edges
  - Use weighted graph
    - Most algorithms naturally allow weights
    - May severely affect community detection algorithm runtime

How to prune edges

- Method 1.2: Majority voting
  - Consider only the consistent scores (omit the outliers)

outlier outlier

2.12 2.35 1.85 2.36 2.53 1.29 2.01 2.32 1.44

All pairwise **inter-class** scores (where each class has 3 images)

0.0 1.39 1.01 1.39 0.0 1.35 1.01 1.35 0.0

All pairwise **intra-class** scores (where each class has 3 images)

  - Many possible ways to do that

How to prune edges

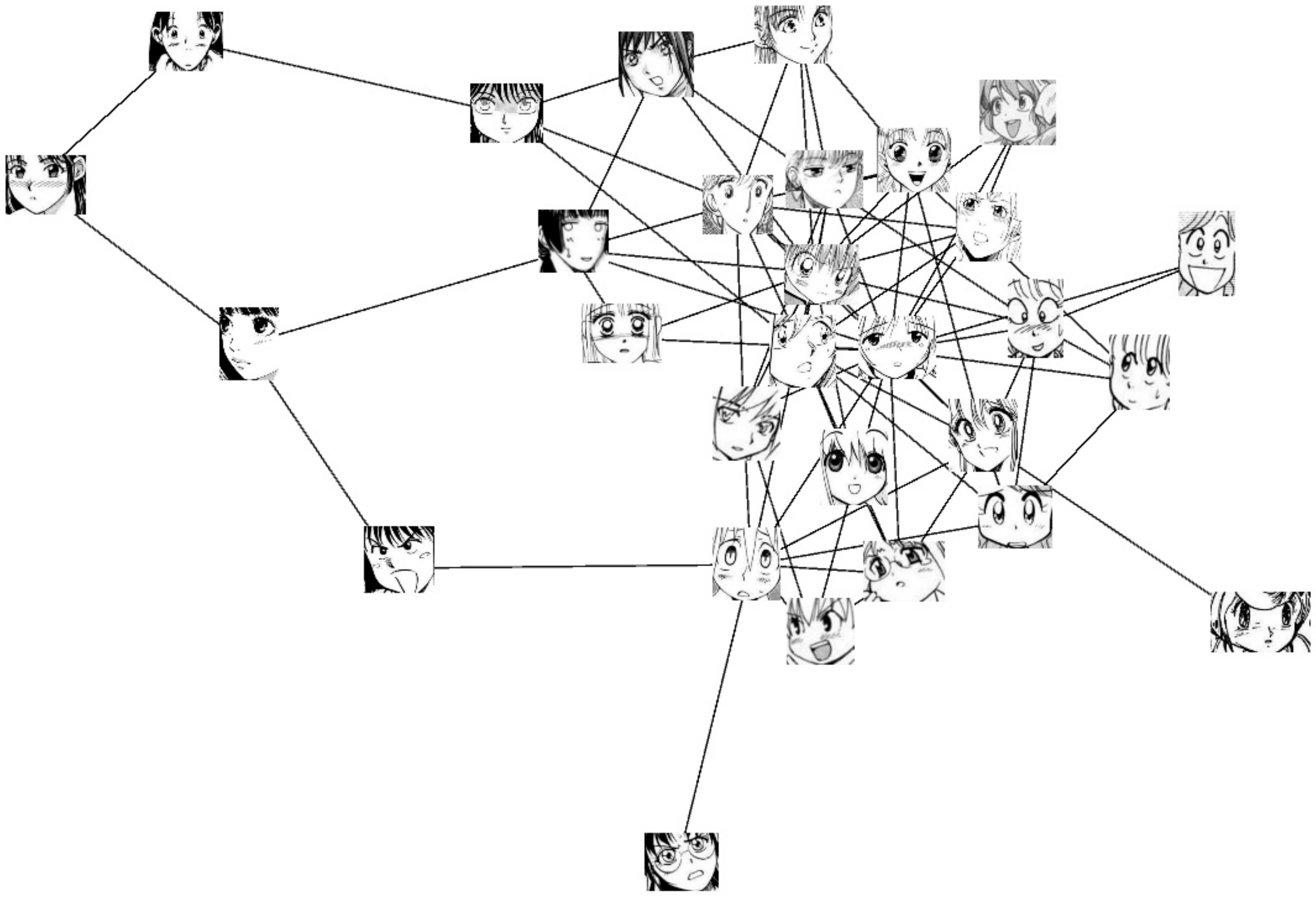
- Method 2: Use all edges
  - Use weighted graph
    - Most algorithms naturally allow weights
    - May severely affect community detection algorithm runtime

Project(s) in the past

- Many projects in the past were based on the Siamese Network
  - Architectural Genres (Data: photos of buildings from internet)
  - Monkey Species (Data: monkey faces from internet)
  - Paintings (Data: WikiArt)
  - Fashion (Data: Fashion run-way photos from internet)
  - City landscape (Data: from internet)
  - James Bond actors (Data: from internet)
  - University badges (Data: Misc)
  - Movie Posters (Data: from internet)
  - Calligraphy (Data: from internet)
  - Game Posters (Data: Steam)
  - See “Files > Projects” in Luminus for details
- How to analyze: Comic book faces

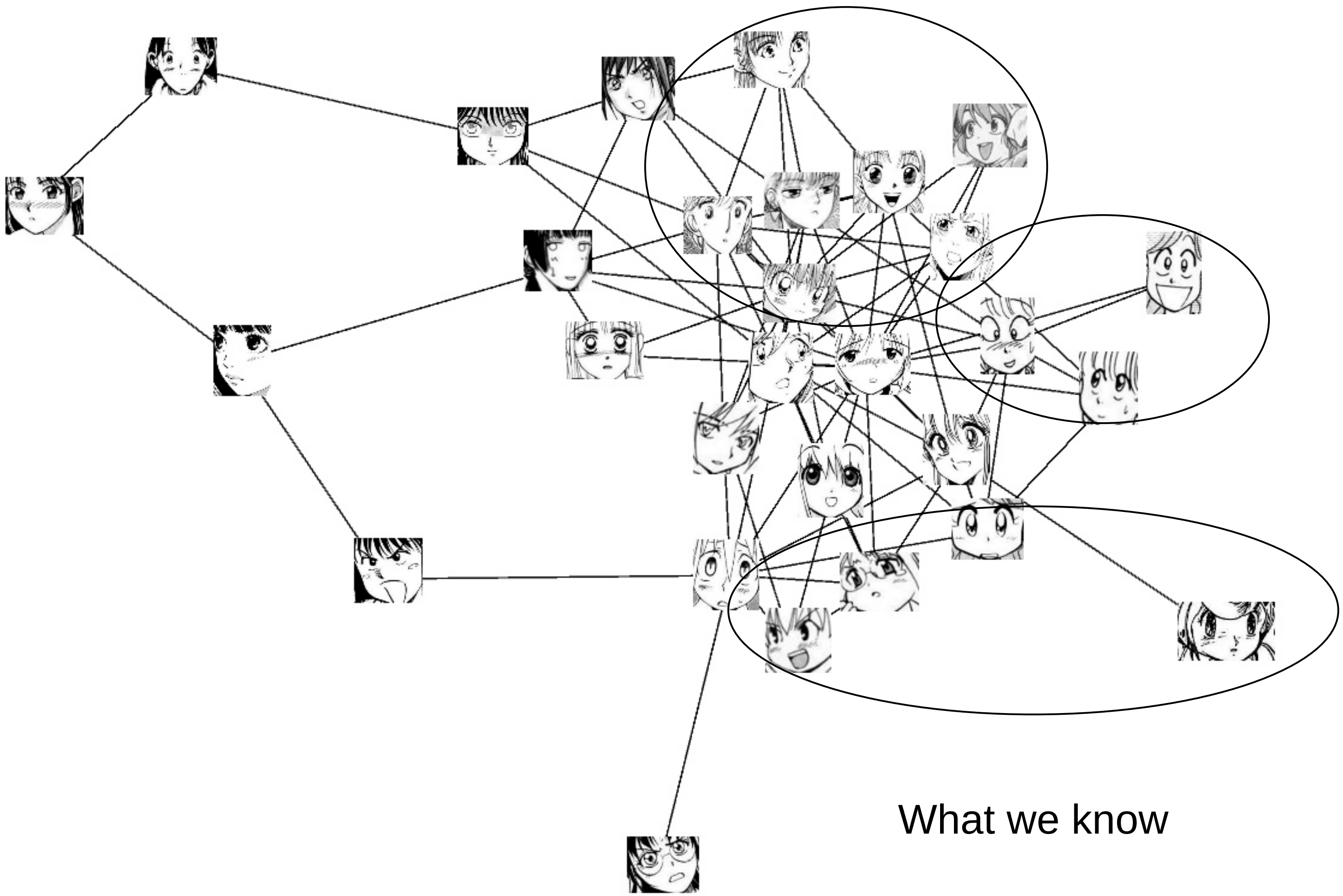
Example from 33 comic books

- Do you agree with this network?



Example from 33 comic books

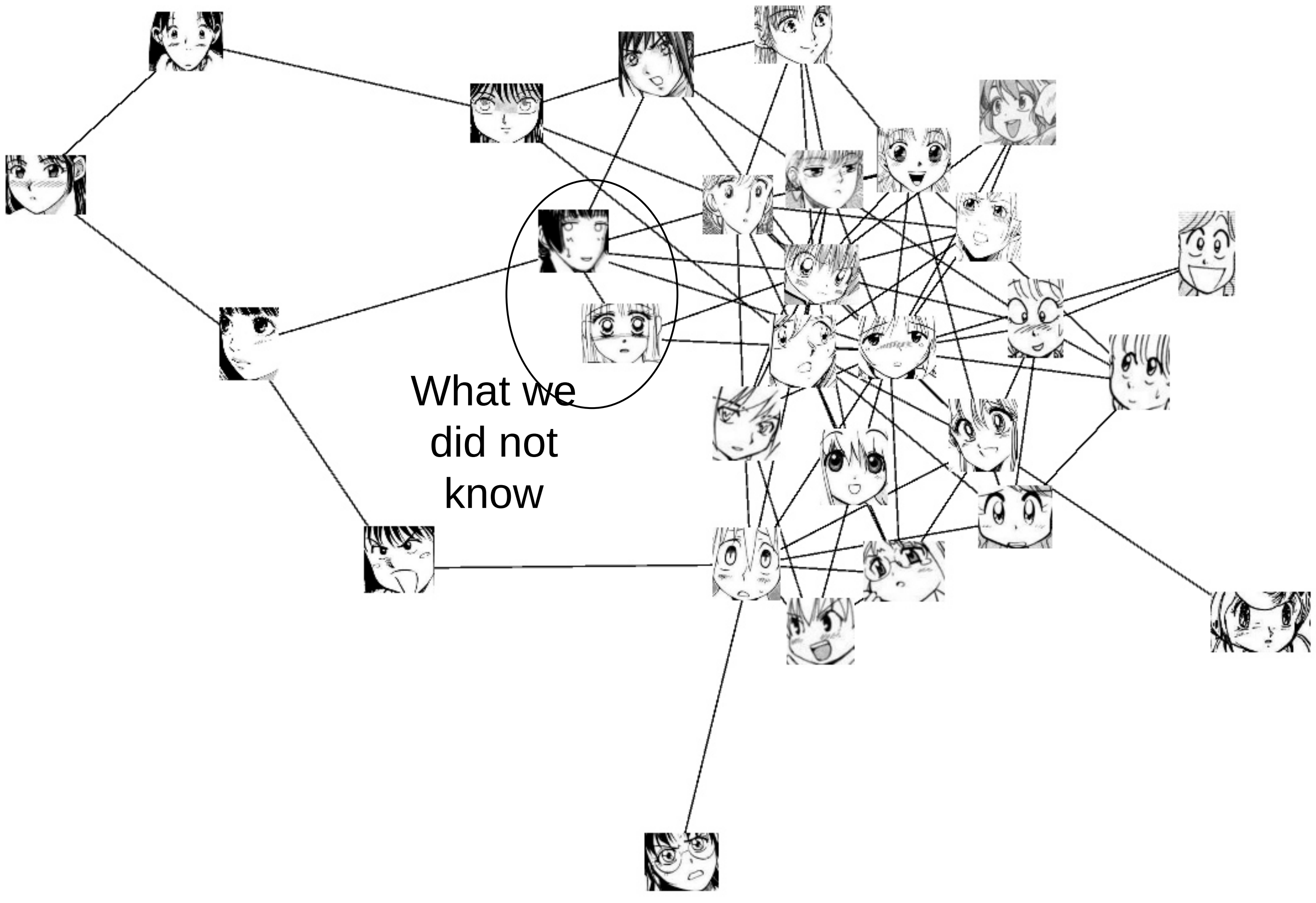
- Do you agree with this network?



What we know

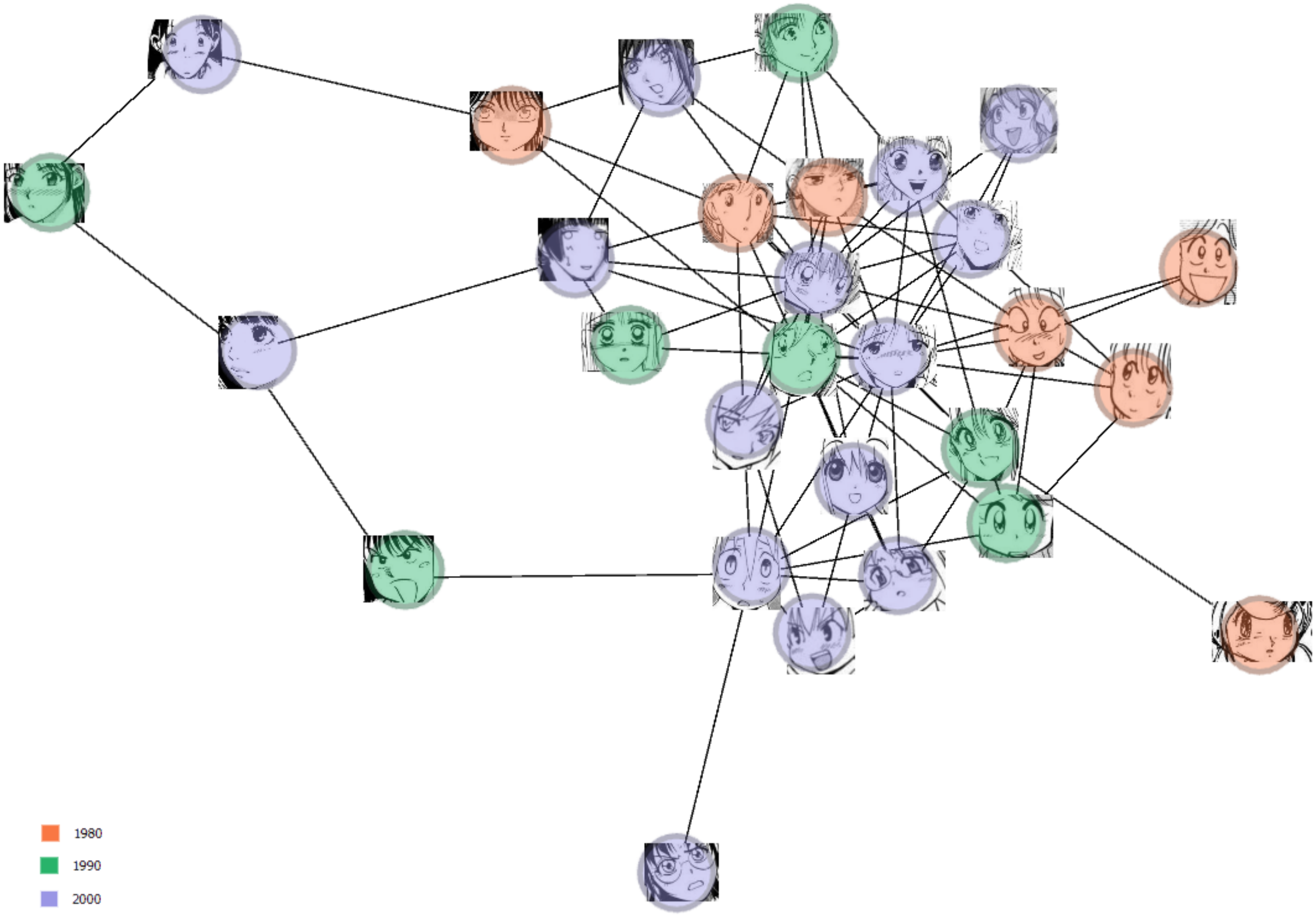
Example from 33 comic books

- Do you agree with this network?



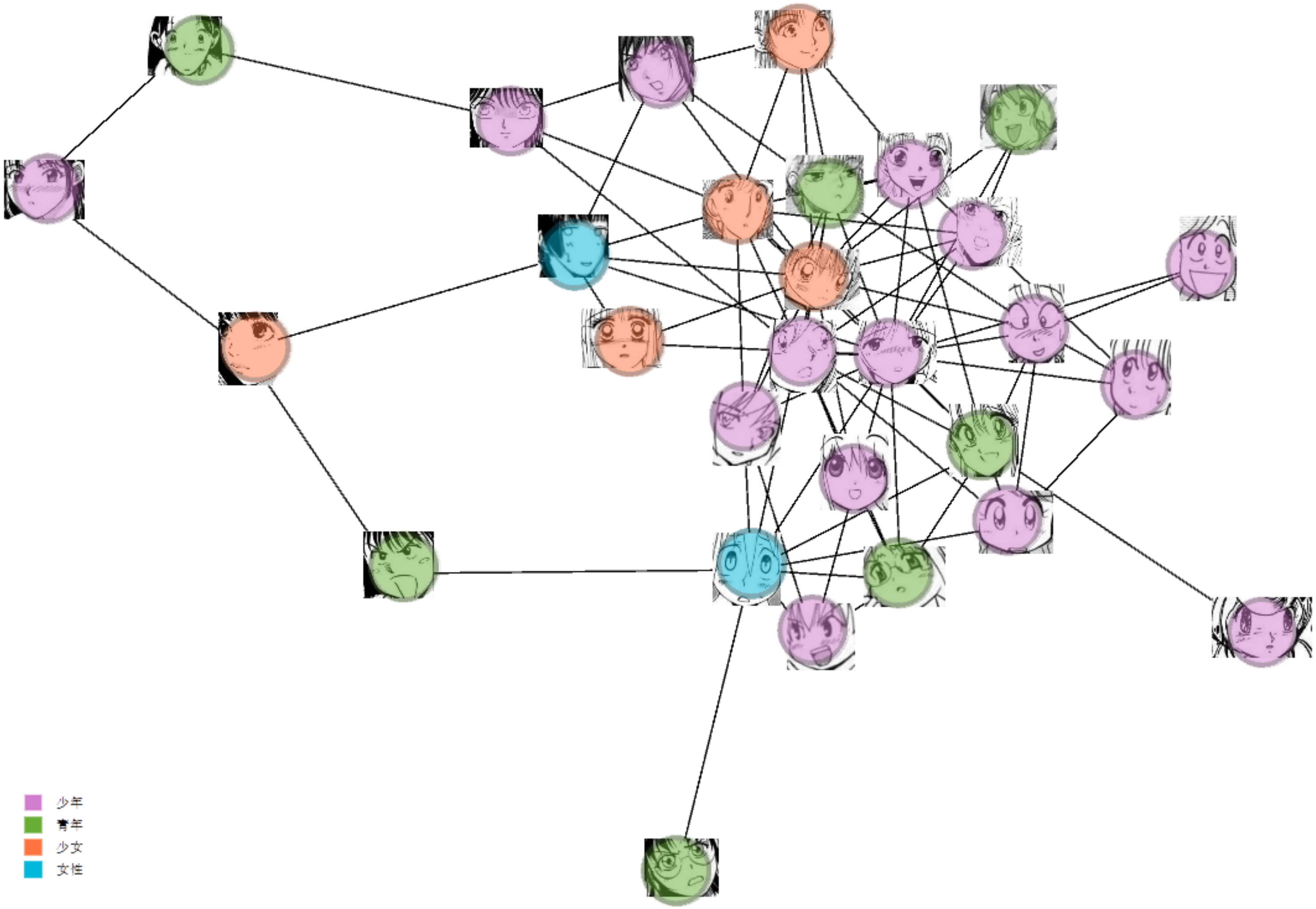
Example from 33 comic books

- The most important part is the analysis



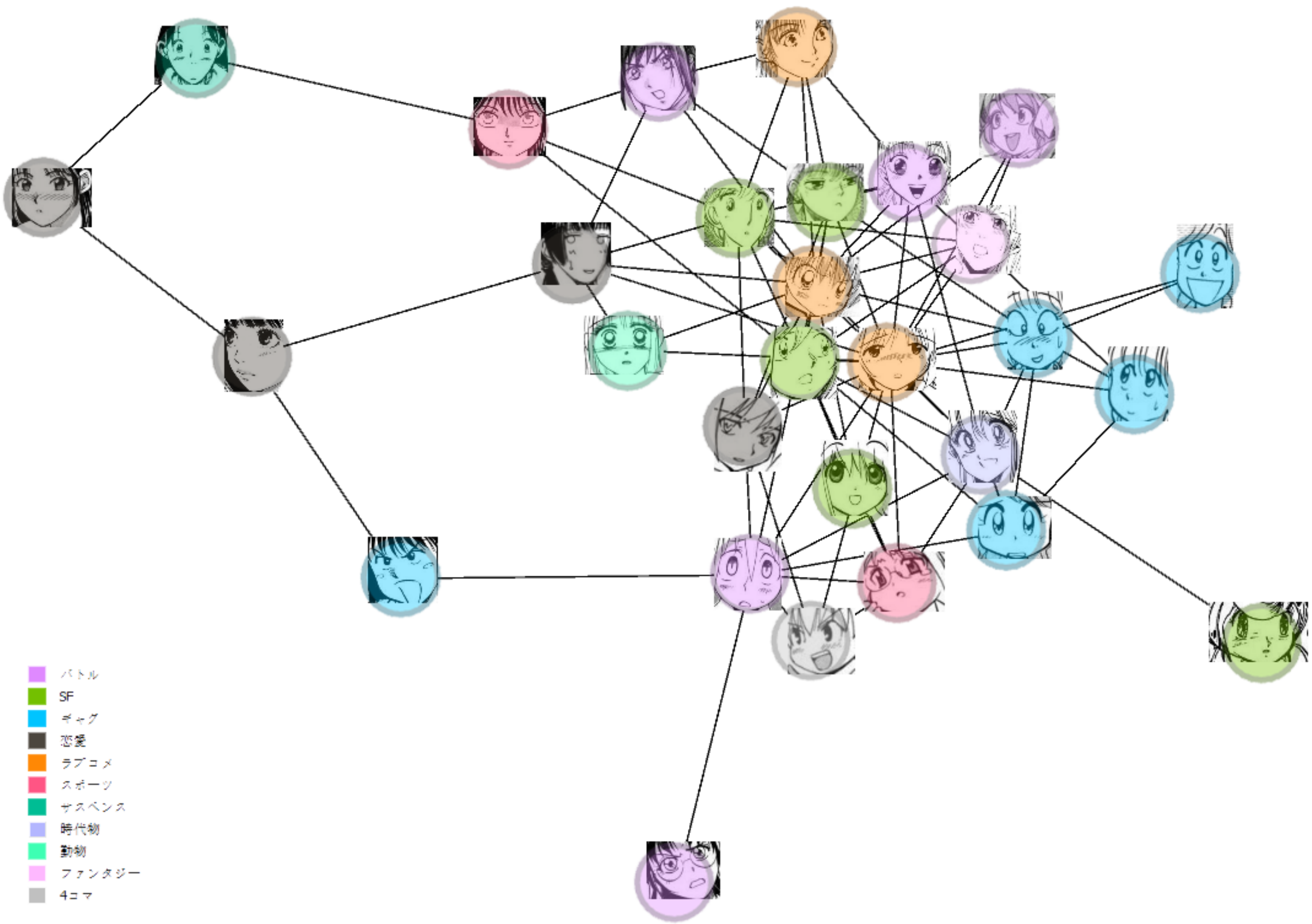
Example from 33 comic books

- The most important part is the analysis



Example from 33 comic books

- The most important part is the analysis

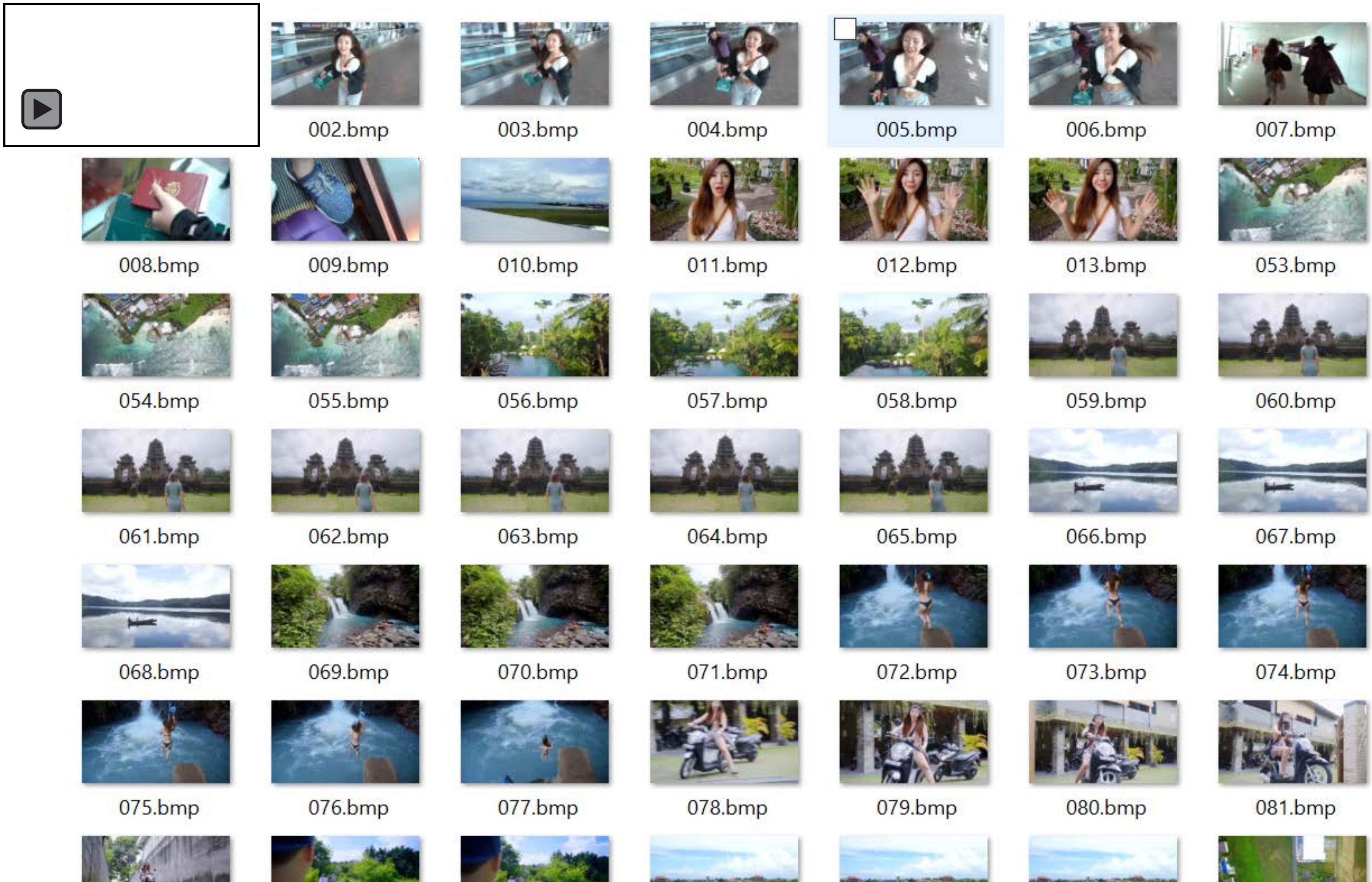


Graph from objects in images

- First identify the elements/objects in the images
  - Can be done with a pre-trained CNN
- Edges can be formed after the objects are identified
  - Use images (or videos) as vertices, add edge between images that share many objects
    - Bag-of-words through places365 (https://github.com/CSAILVision/places365)
  - Use objects as vertices, add edge between objects that appear together in many images
    - Character co-appearance

Bag-of-words through places365

- Sample frames from travel vlog (through maybe ffmpeg)



Bag-of-words through places365

- Identify scenes with places365 (https://github.com/CSAILVision/places365)

Bag-of-words through places365

- Identify scenes with places365 (https://github.com/CSAILVision/places365)

Wordcloud visualization of scenes obtained with places365

- The scenes obtained from places365 can be used as a bag-of-word
- The number of matches between the bag-of-words of two videos gives a similarity score between the videos
- Up to your imagination

Project(s) in the past

- Only one project used this method in the past
- Music genre analysis
  - 173 music videos from YouTube
  - Screenshots taken according to fixed time interval
  - Places365-CNN used to identify the background
  - Discovery
    - Music of different genre has preference for background
  - See “Files > Projects” in Luminus for details

Character co-appearance

- https://www.reddit.com/r/dataisbeautiful/comments/8e3eax/marvel_cinematic_universe_20082018_preinfinity/dxs1imj/

Scene2Links OC: 1 481 points · 3 months ago

I created a graph network of the Marvel Cinematic Universe. The data source is the movies themselves, I compiled the information myself. Data was visualized using Gephi and output as a SVG and made into an image using Inkscape. Each node (circle) represents a character, while the edges (lines) between circles represent character interactions. Character interaction interactions include speaking, fighting, viewing film/footage, electronic communication, and flashbacks/visions/dreams). The more screen time (in minutes) two characters share, the thicker the edges.

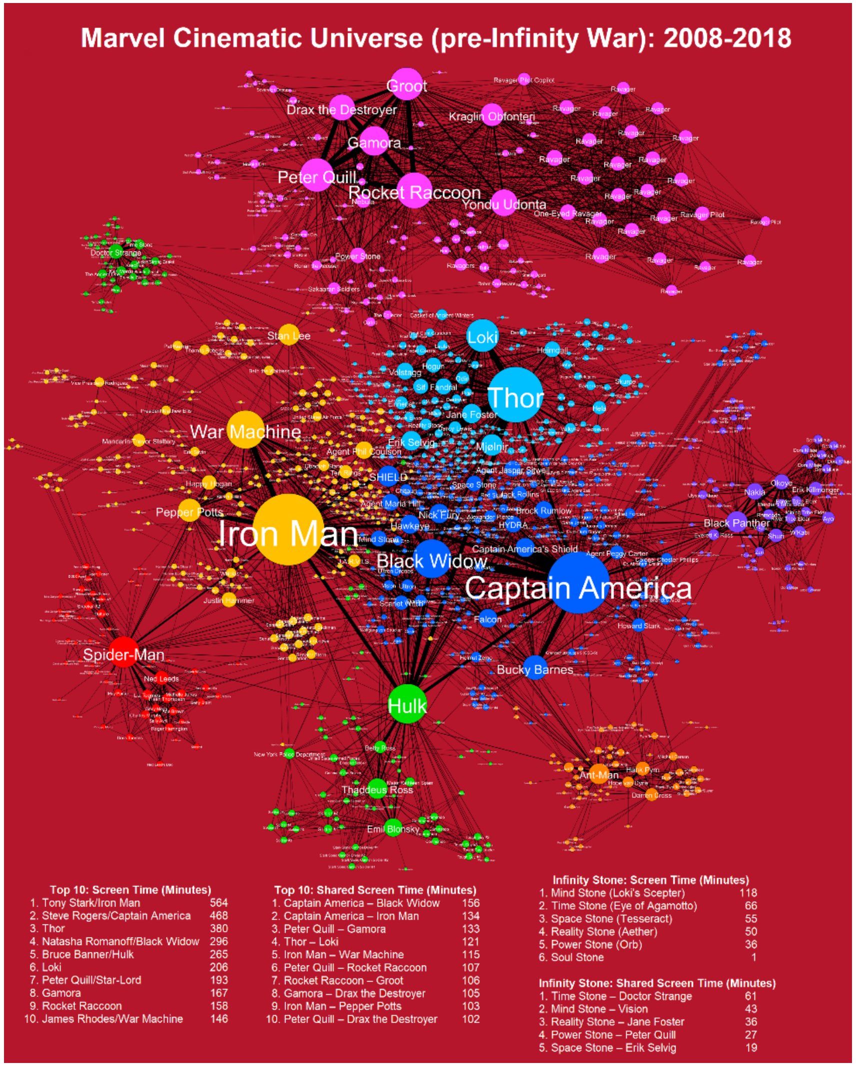
In addition to characters, I included a few objects like Captain America's Shield and Thor's hammer, Mjølnir; however, for the most part, objects outside of the Thor films were ignored. In some cases, henchmen and unidentifiable CGI characters were combined into one node (e.g., HYDRA soldiers are simply labeled HYDRA). I have also included nodes for the Infinity Stones, which you can find if you hunt through the connections in the network.

The color of the nodes denote communities, which represent nodes that are more interconnected to each other than to the rest of the network. Quite naturally, characters fall into communities associated with their films. In this analysis, characters fell into 9 communities. While some may consider there to 10 separate groups (Iron Man, Hulk, Thor, Captain America, Avengers, Guardians of the Galaxy, Ant-Man, Doctor Strange, Spider-Man, and Black Panther), given the heavy interactions between SHIELD and Captain America, the algorithm classifies Avengers and Captain America films as one community.

Share Save

Character co-appearance

- https://www.reddit.com/r/dataisbeautiful/comments/8e3eax/marvel_cinematic_universe_20082018_preinfinity/dxs1imj/

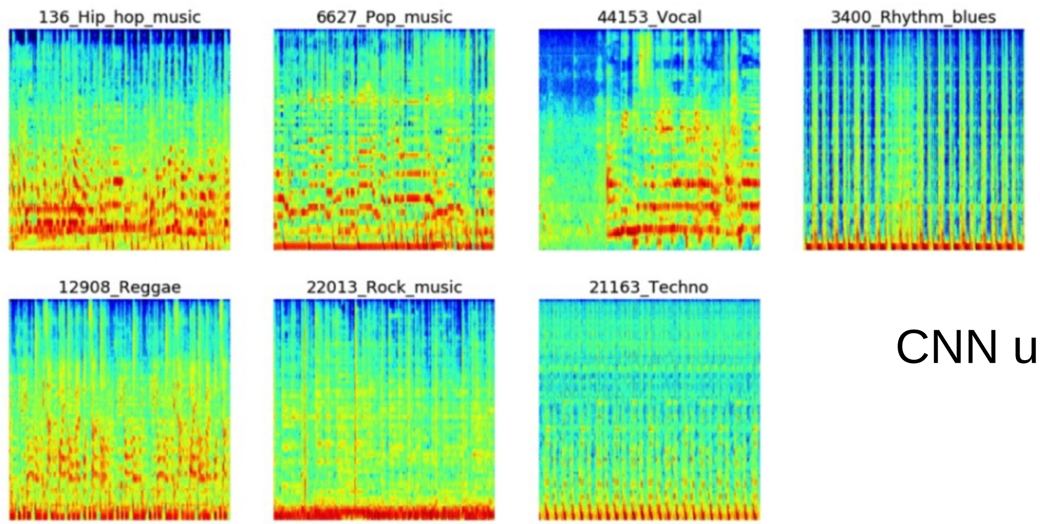


Project(s) in the past

- None so far

Graph from spectrograms

- What is the extent of using spectrogram images to distinguish sounds?
  - Can spectrogram differentiate music genre?
    - Music Genre Classification using Machine Learning Techniques https://arxiv.org/abs/1804.01149

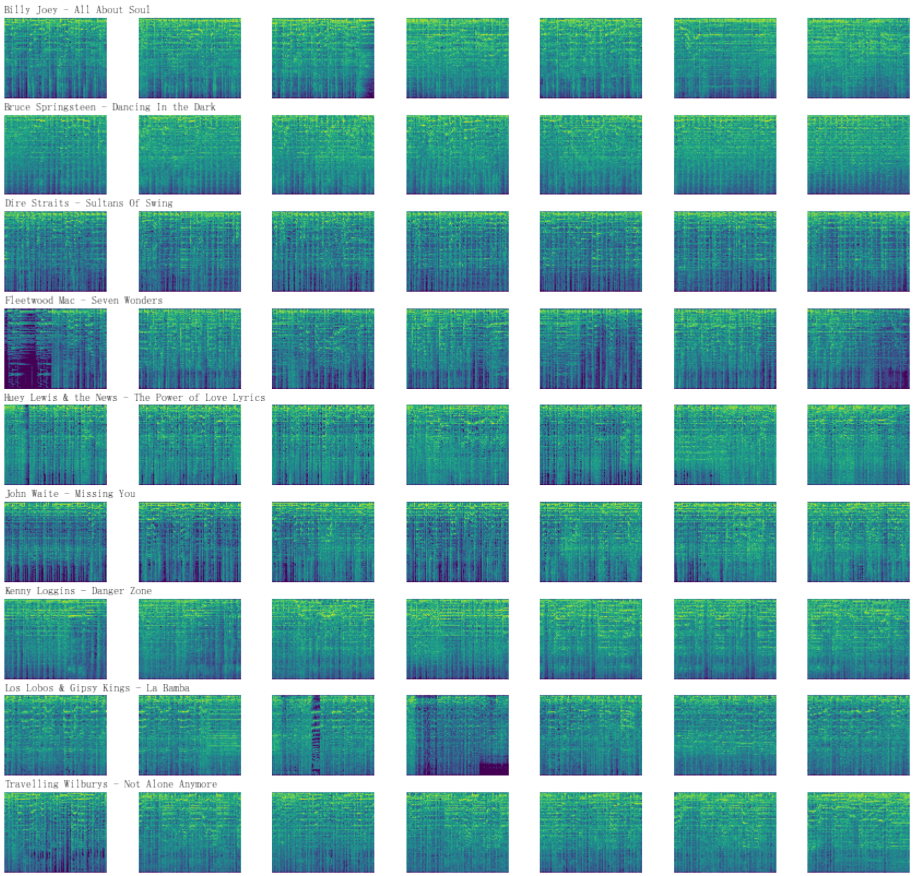


CNN used for classification

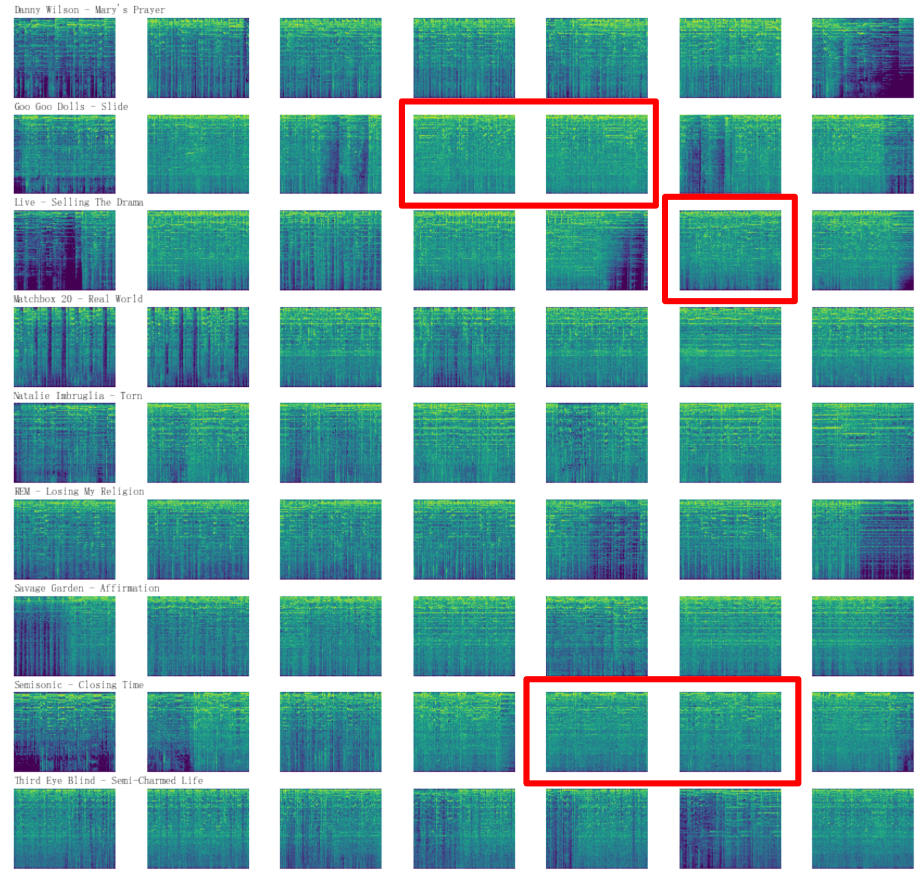
    - Oversized CNNs may be able to overfit to identify subtle cues in the images, but our simplistic Siamese Network may not
- Let's take a closer look at the spectrograms

Spectrograms of different genres

Classic Rock & Roll



Alternative Rock

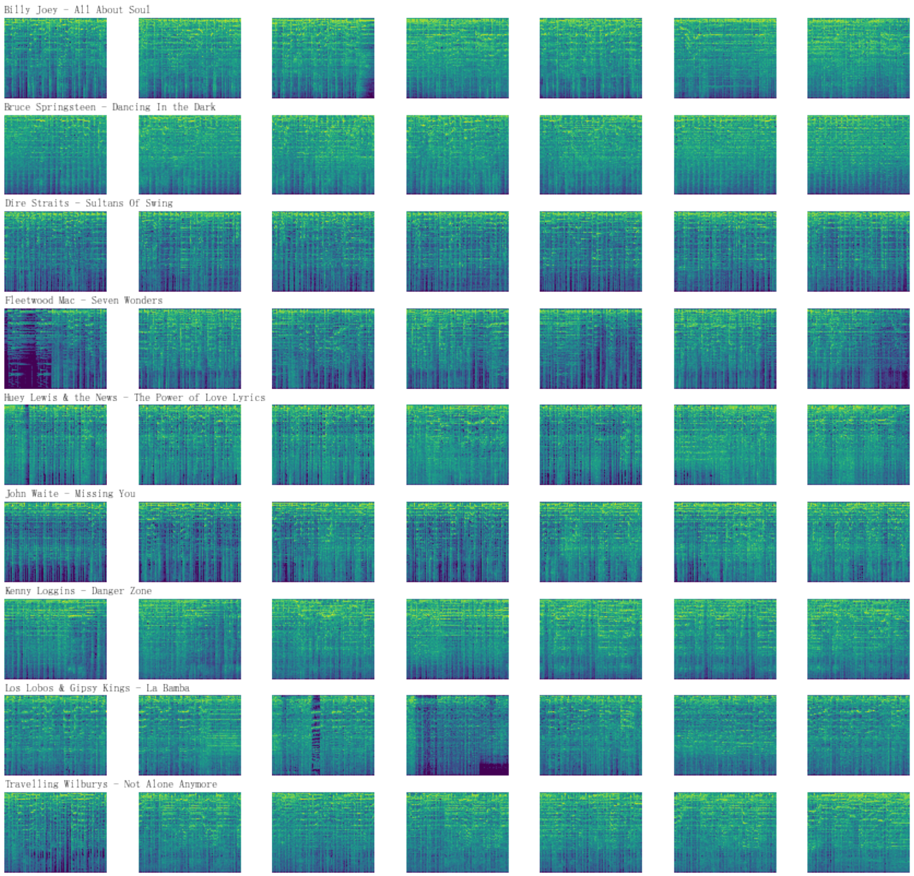


Not much difference between classic Rock & Roll (80s) and Alternative Rock (90s)

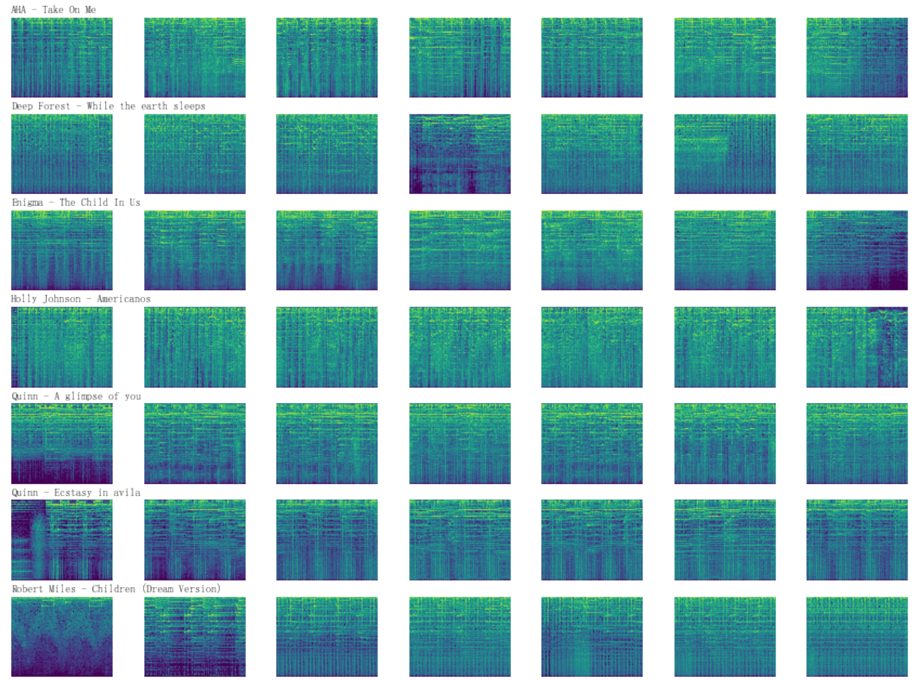
- Occasionally some parts of the latter are very noticeably faster and louder

Spectrograms of different genres

Classic Rock & Roll



Electronic dance music

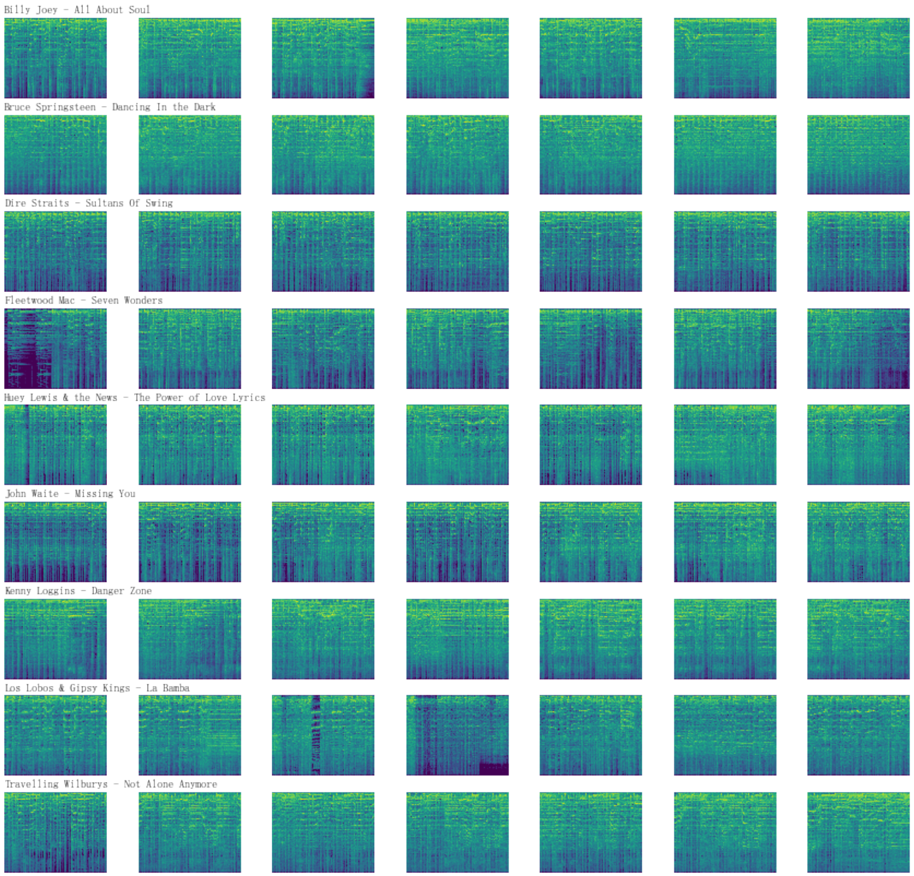


Classic Rock & Roll (80s) is on par with electronic dance music

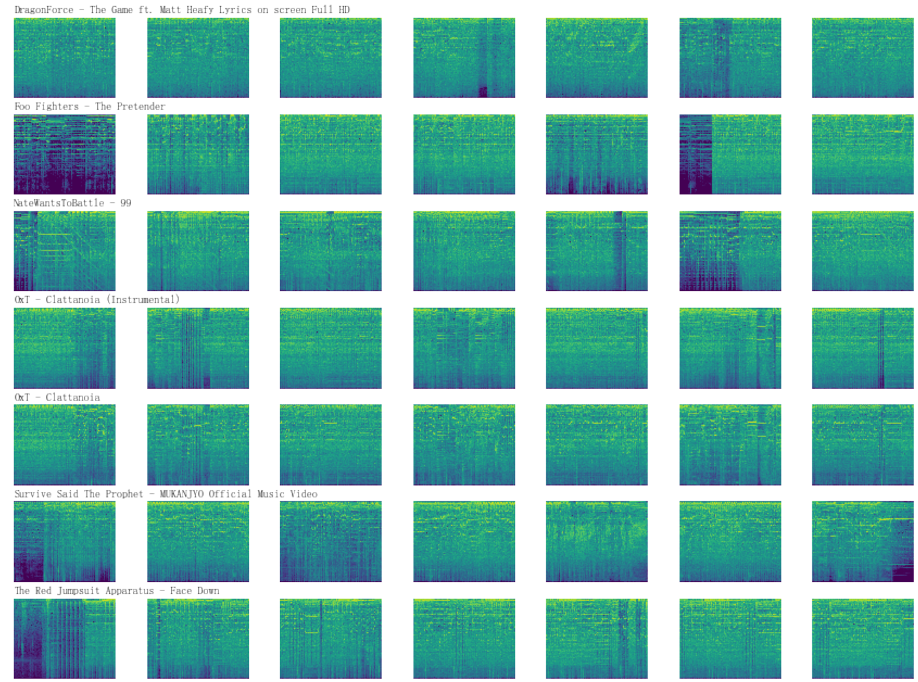
- Occasionally some parts of the latter are very noticeably faster and louder

Spectrograms of different genres

Classic Rock & Roll



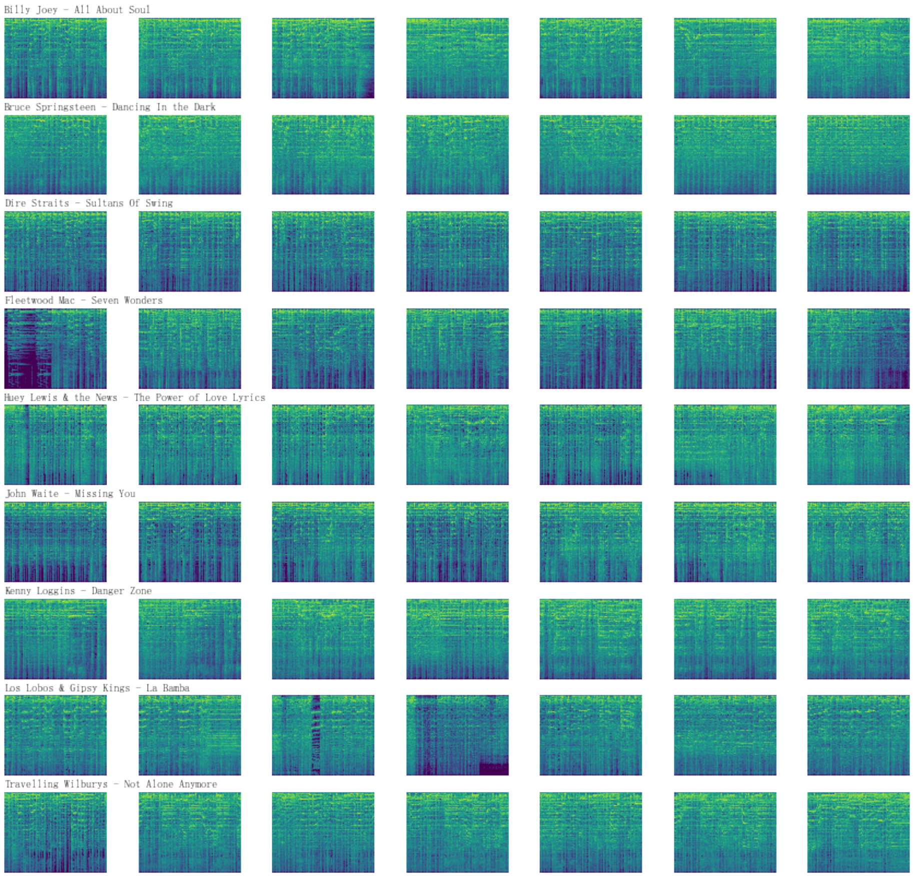
Grunge/Thrash



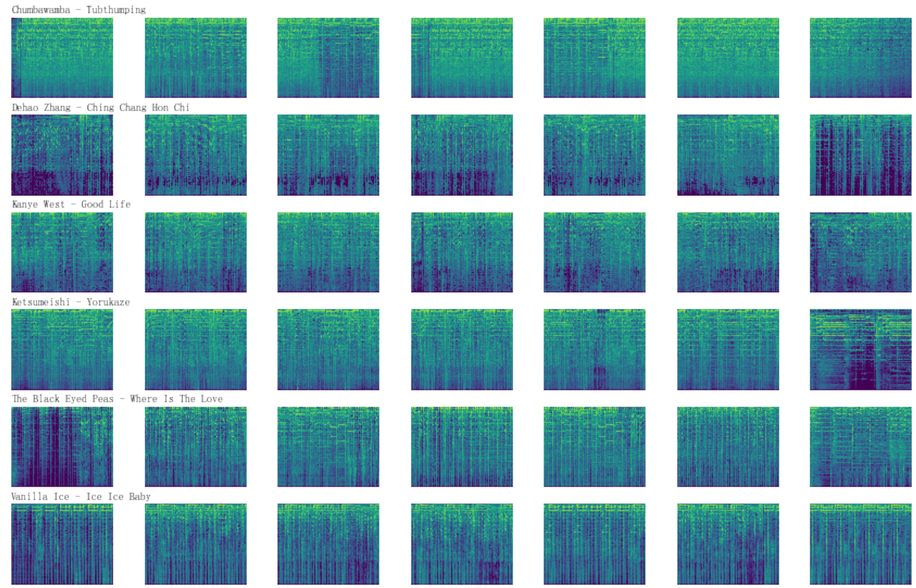
Grunge/Thrash (21st century) is much faster and “fuller” than classic Rock & Roll (80s)

Spectrograms of different genres

Classic Rock & Roll



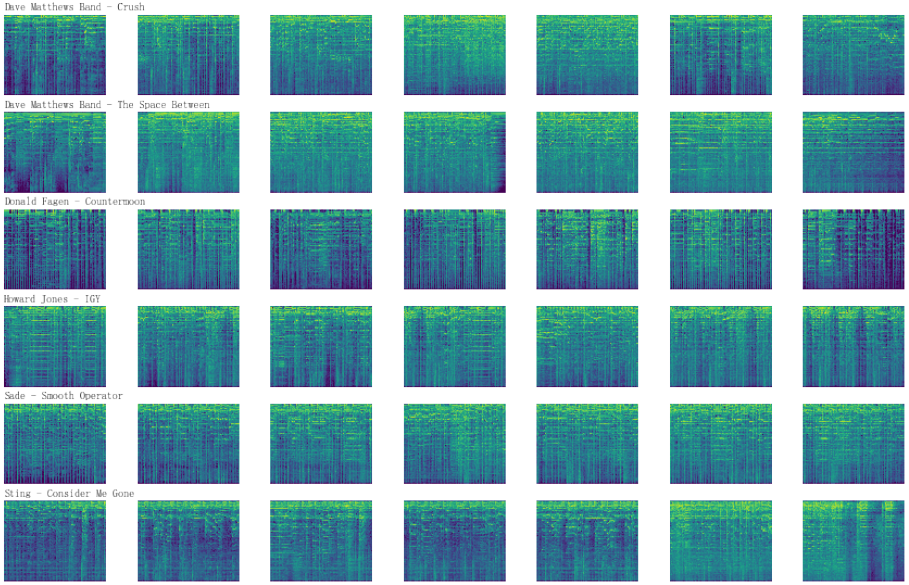
Hip-Hop



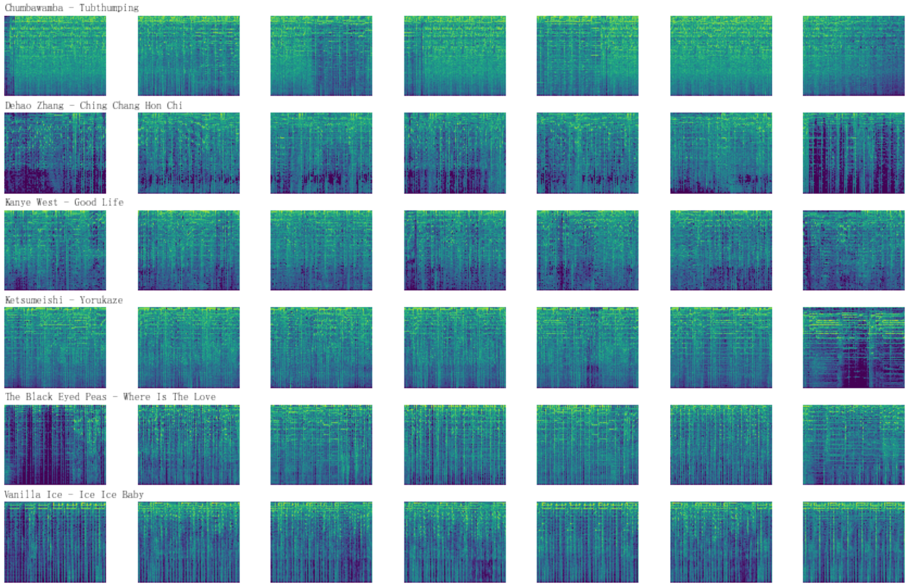
Hip-Hop tends to be slower paced than classic Rock & Roll

Spectrograms of different genres

Jazz-pop



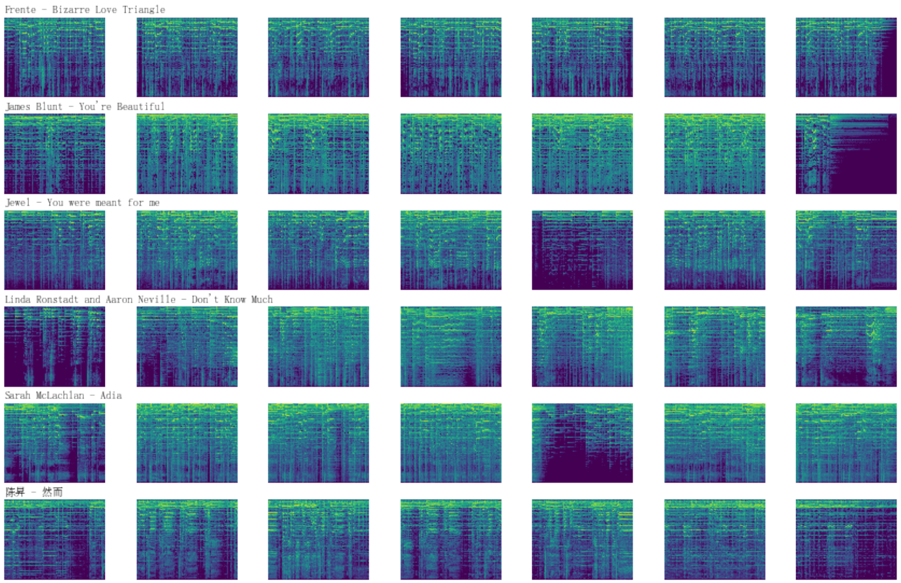
Hip-Hop



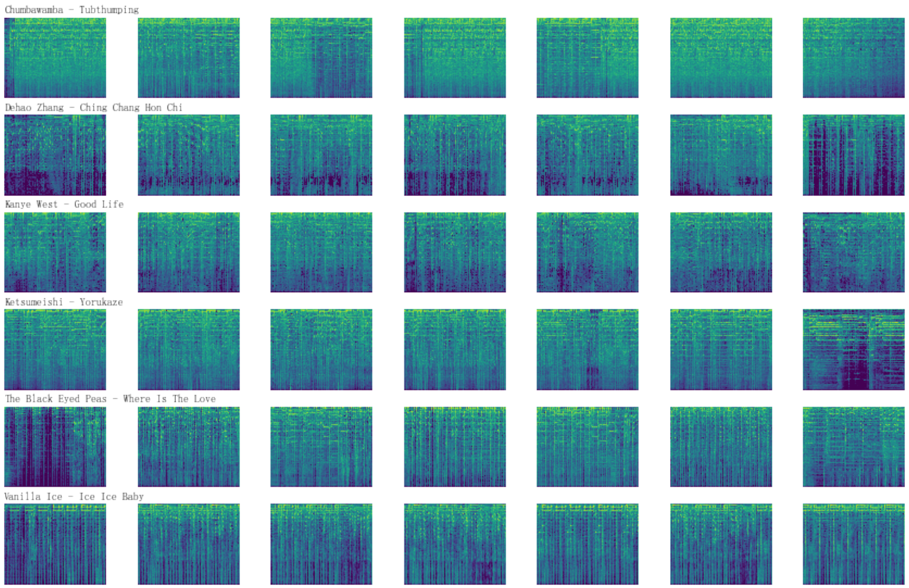
But Hip-Hop is not that different from Jazz-Pop

Spectrograms of different genres

Slow acoustic



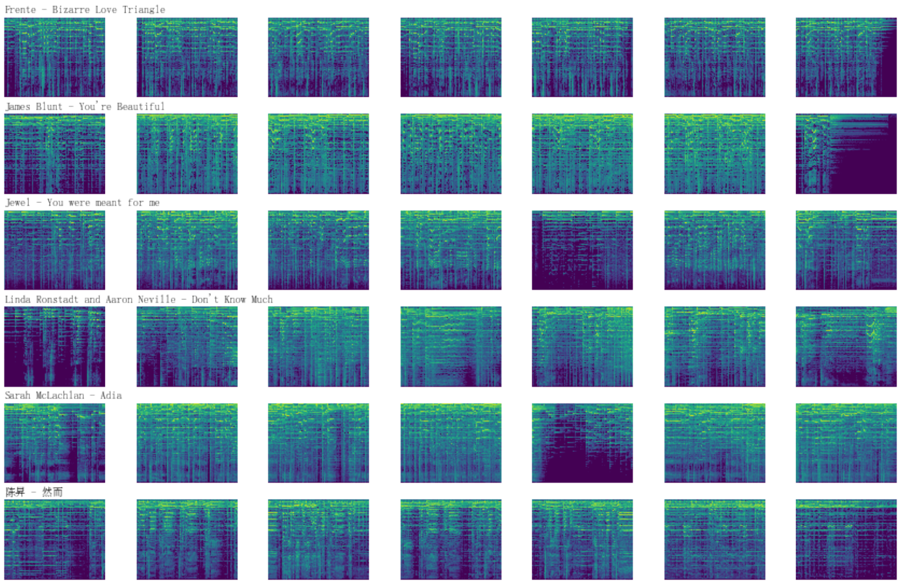
Hip-Hop



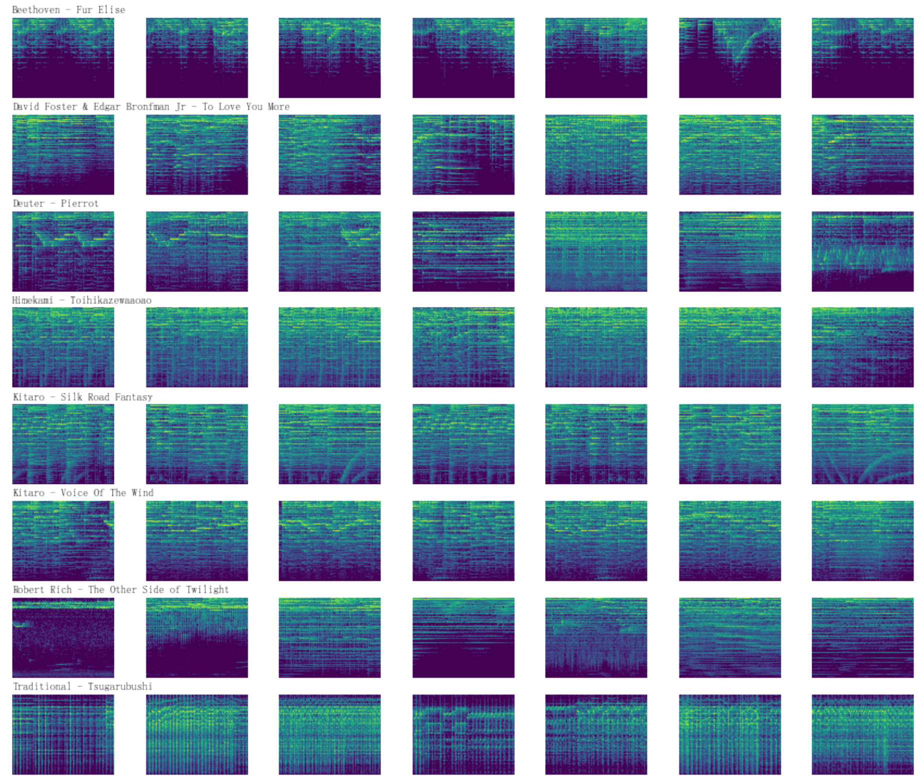
Still Hip-Hop has more energy than slow acoustic music

Spectrograms of different genres

Slow acoustic



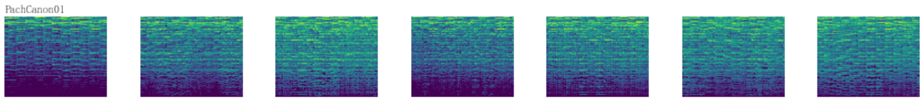
Instrumental



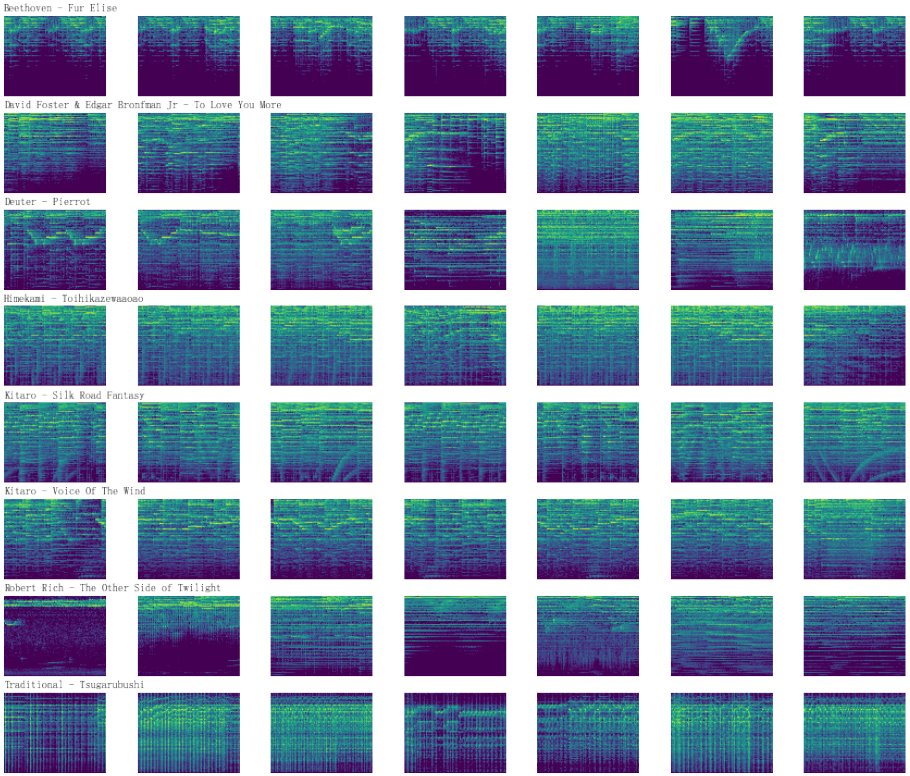
Of course, we can see more variations in instrumental music (e.g. classical) than in pop songs

Spectrograms of different genres

Pachelbel Canon



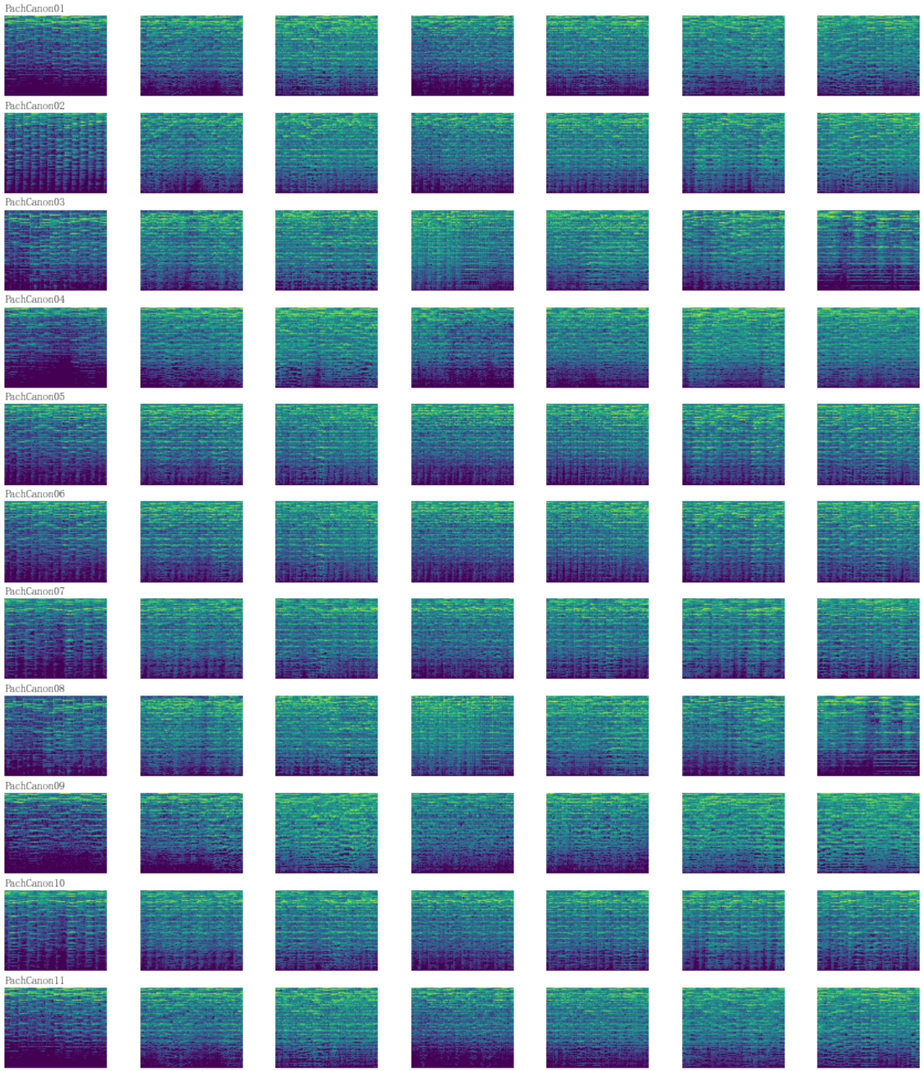
Instrumental



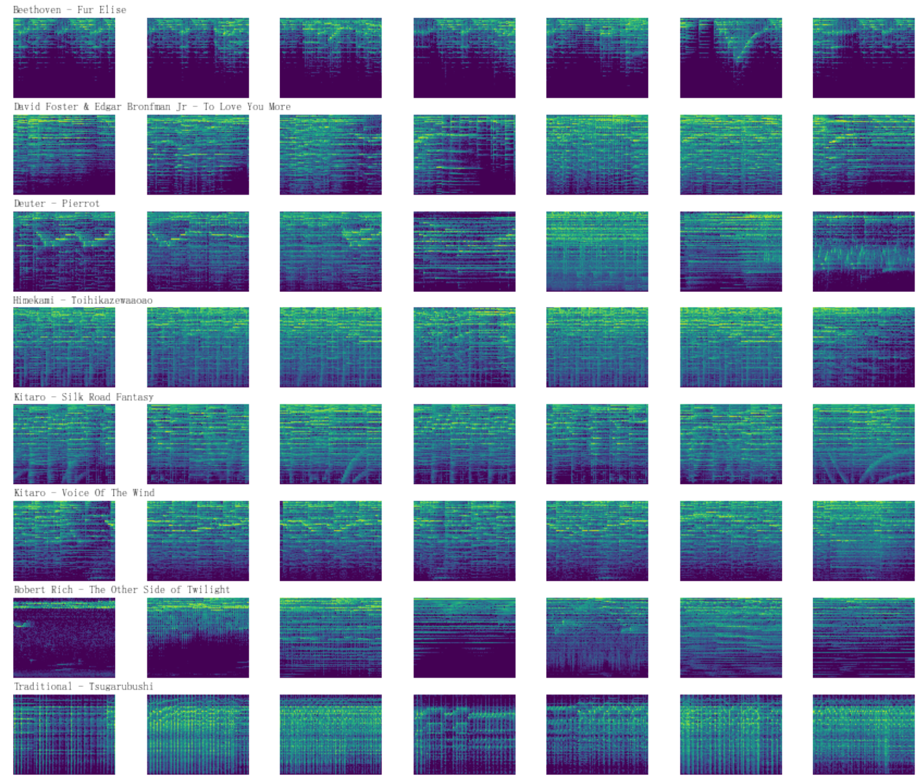
…with some exceptions

Spectrograms of different genres

Pachelbel Canon



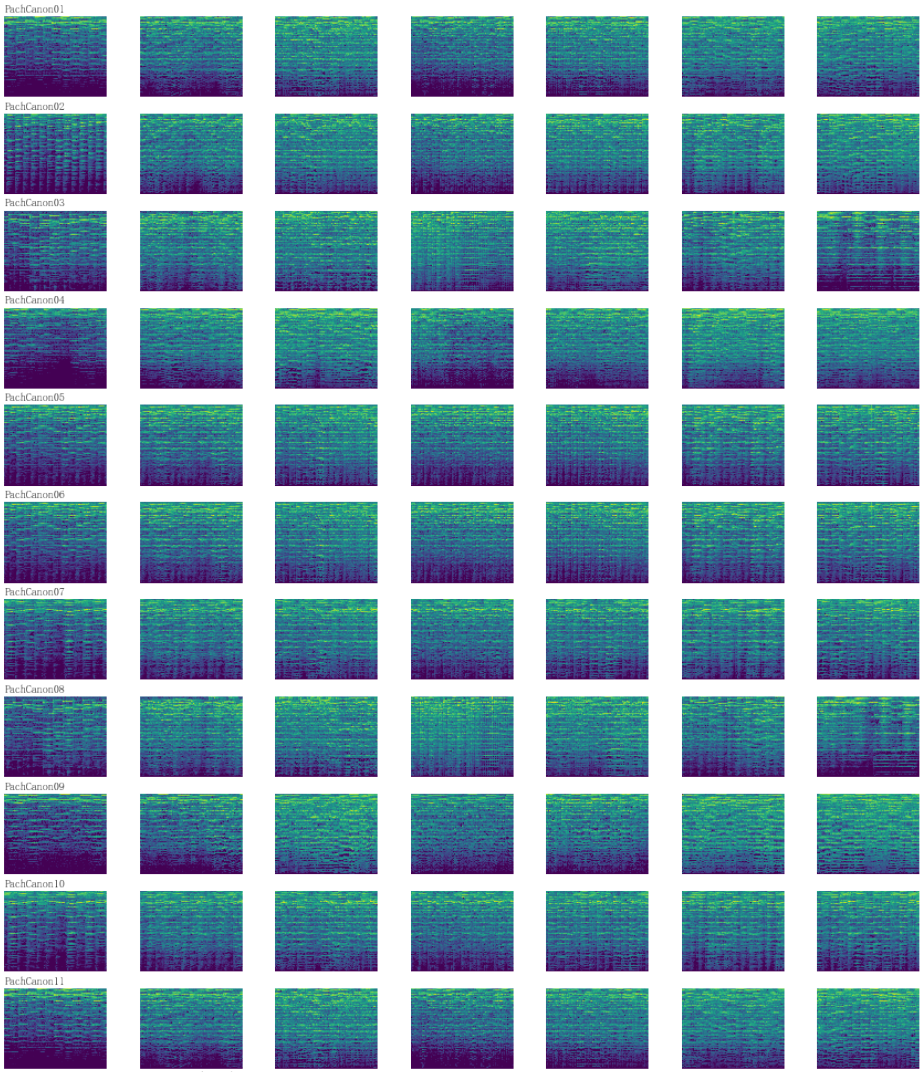
Instrumental



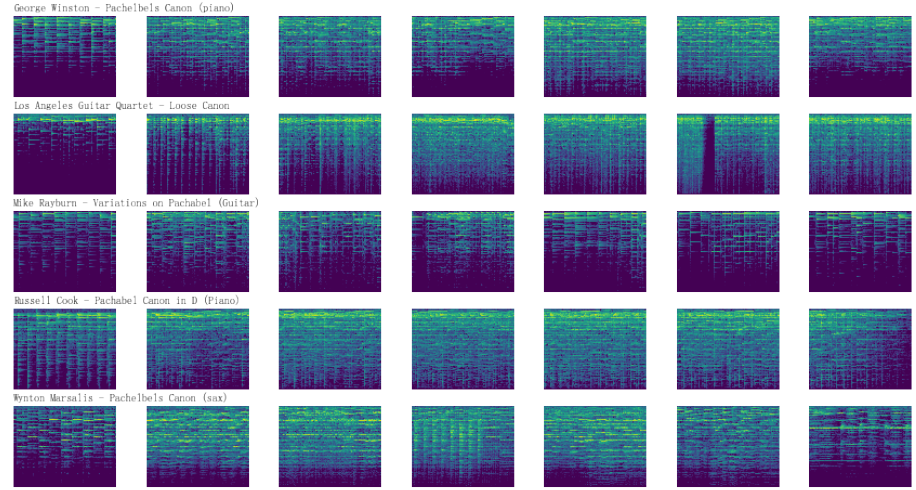
All the Pachelbel Canons are the same!

Spectrograms of different genres

Pachelbel Canon



Pachelbel Canon (rearrangements)



Rearranged canons using other instruments give different spectrograms

Spectrograms of different genres

Case Study 1: Variations on Theme of One Punch Man

Acoustic guitar, Drum

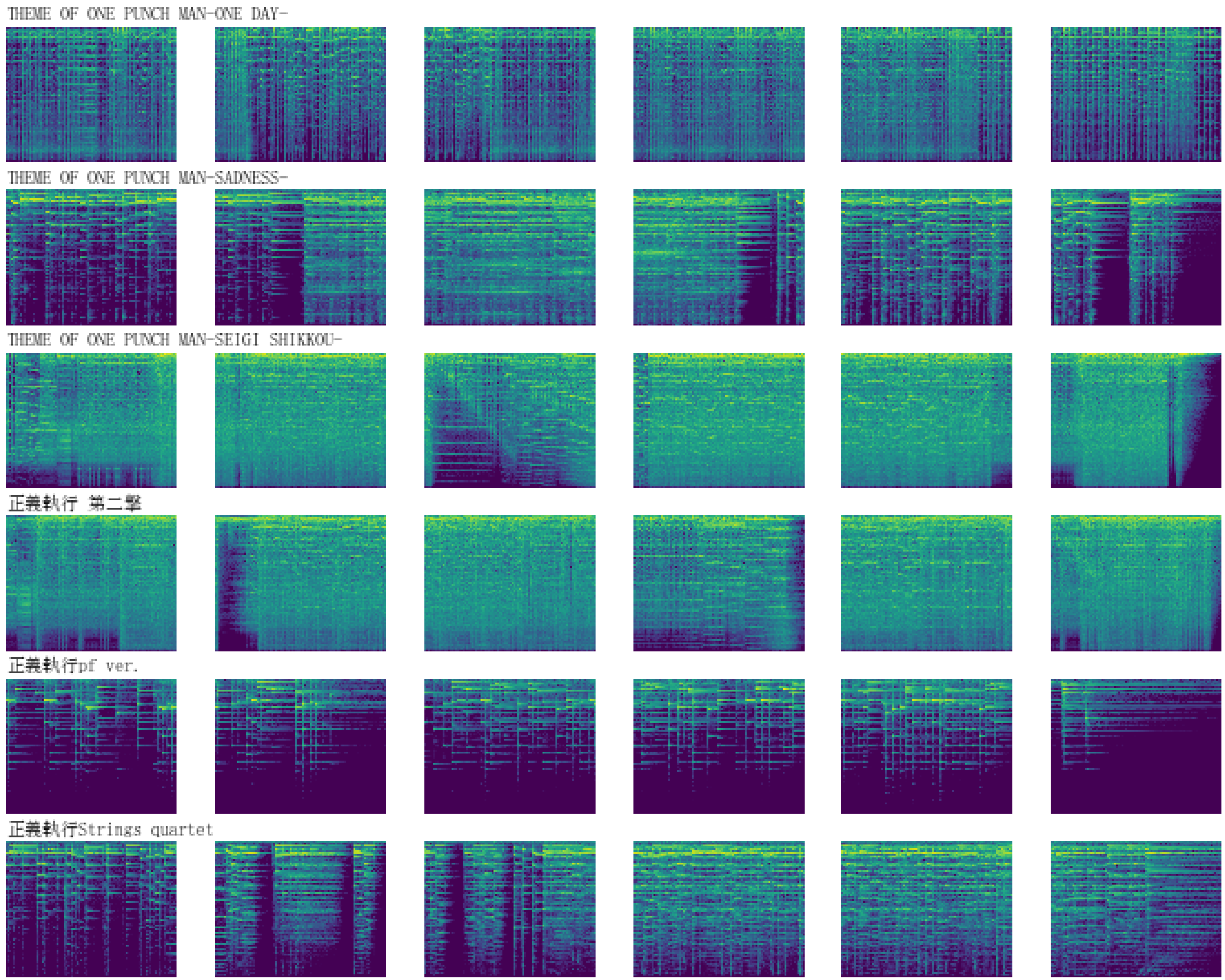
Acoustic guitar

Misc. electric guitars, Drum, Violin

Electric guitar, Drum, Horns, Violin

Piano

Acoustic guitars, Violin

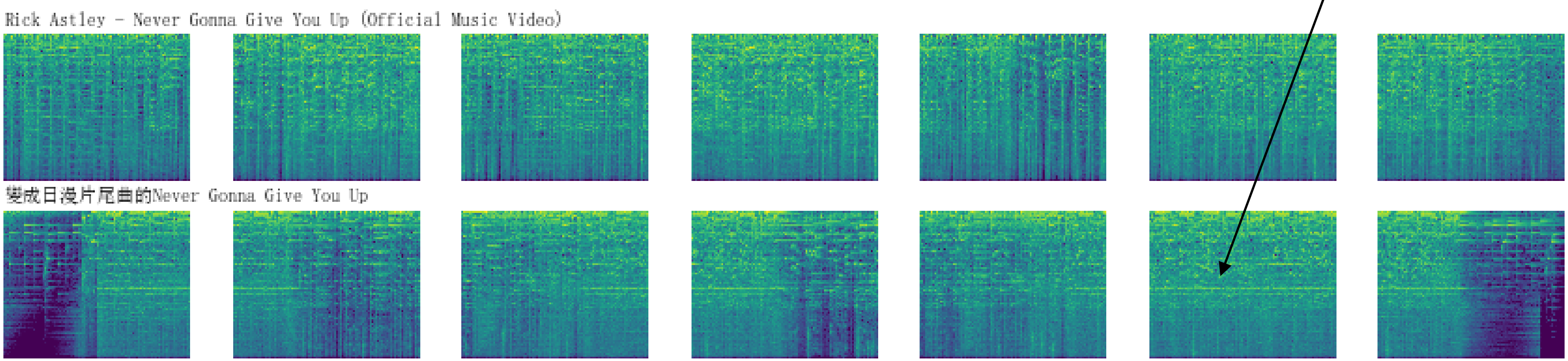


Spectrograms of different genres

Case Study 2: Never Gonna Give You Up

Notice the stronger horizontal lines

Original: More stable rhythm and fewer instruments

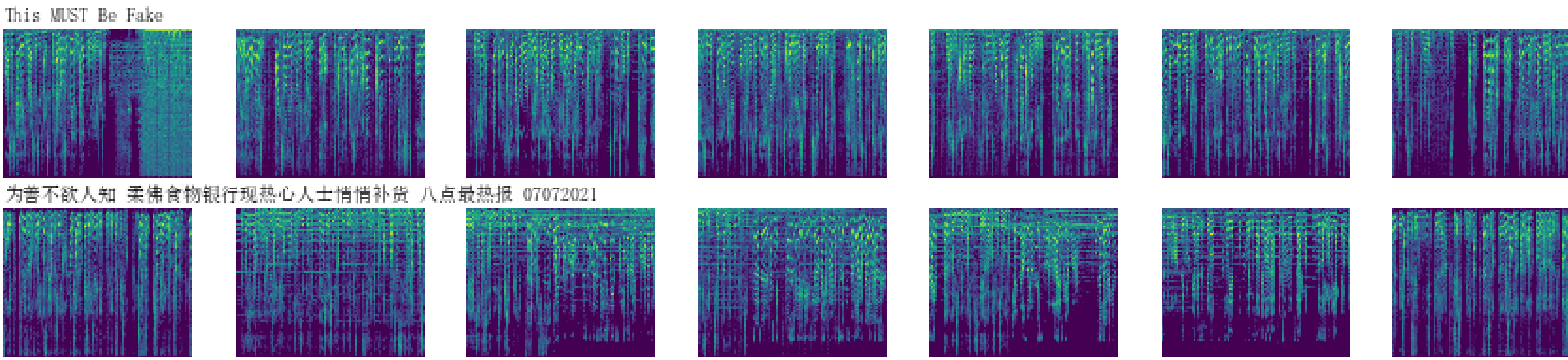


Anisong version: Faster vocal (also auto-tuned?), more tension build-up and more ornaments

Spectrograms of different genres

Case Study 3: Plain speech

English

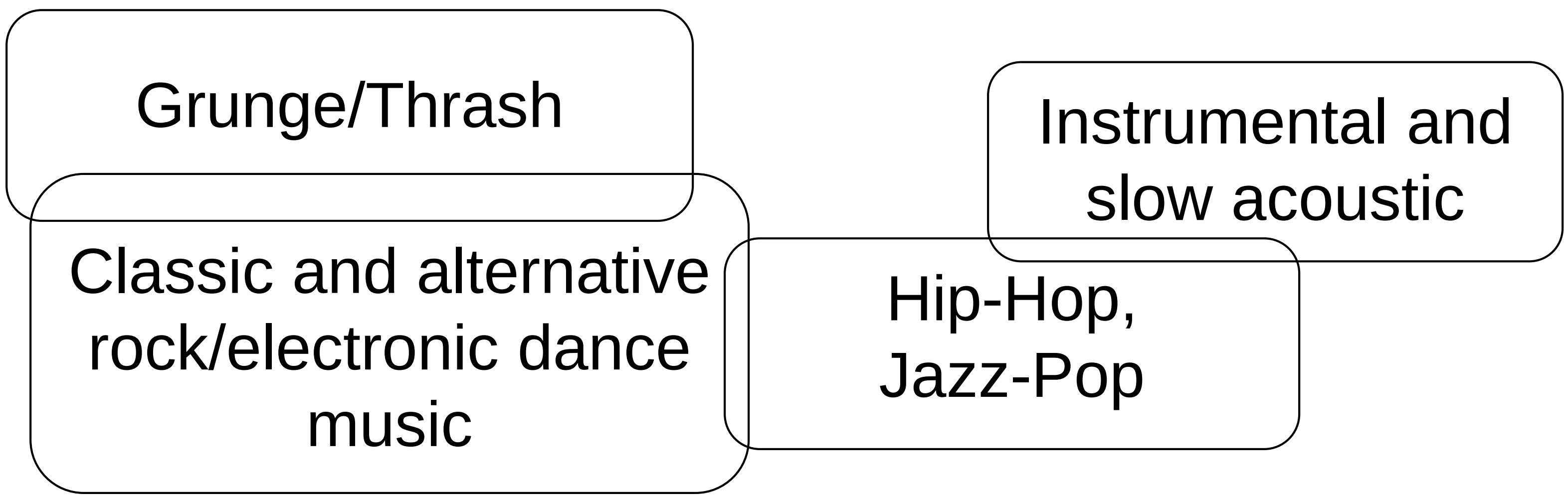


Mandarin

Needs more data to tell whether it is possible to distinguish between speech of different languages

Conclusion

- Music genres are too loosely defined such that classification would be difficult
  - Clear distinction observed only between these groups



- Advice: Check out a few spectrograms before you decide on what you want to do

Project(s) in the past

- None so far since this is new for this year