Lecture 11 – Motion

Today's Learning Objectives:

1. Describe temporal aliasing and how it creates the wagon-wheel effect.

2. Describe the correspondence effect and how it impacts animation.

Final Project Information: Topic Choice

- Must have final project topic chosen and approved by me on **April 3 by 5 pm**

- Acceptable topics:
  - **A presentation on your research.** You must create a poster with original data visualizations and be ready to defend and explain why you chose to use the visualizations that you did.
  - **A presentation on someone else's data set.** Could be inside or outside CADS. Same rules as above apply. Data set can be from the internet on a topic you find interesting.
  - **Create and present on an R package** that implements a technique not currently available in R. Must be in R (sorry, no python). If something else exists, must be a substantial improvement.

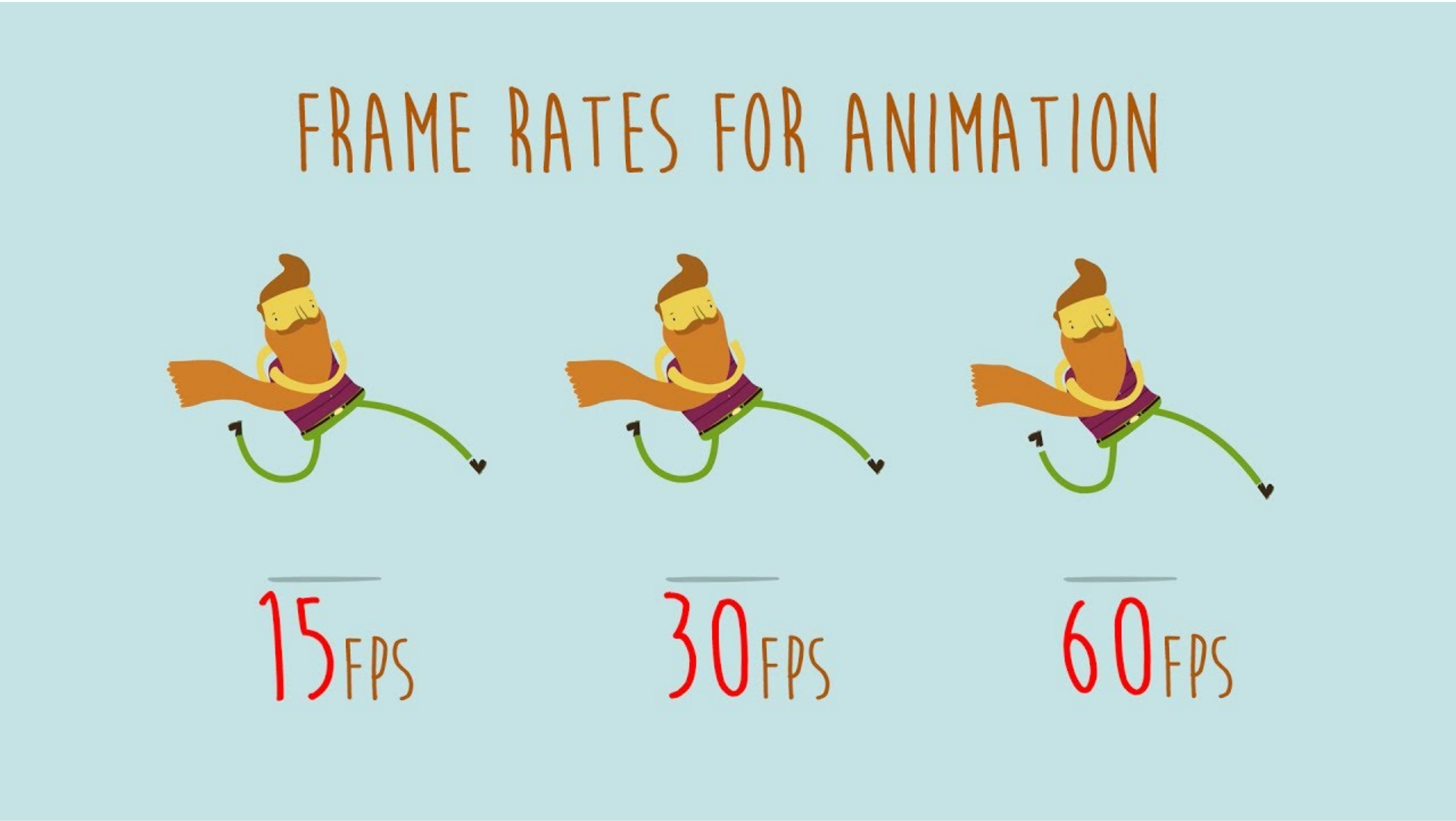
Patterns in Motion

- How are animations created?
  - Computers render individual still images at a ‘frame rate’
  - Frame rate has to be at least 12 frames per second to look like animation is in motion.
  - Faster than 60 fps is completely indistinguishable from smooth motion.

Animation Frame Rates:

https://www.youtube.com/watch?v=npMreLeVD6o

Frame Rates in Animation



Patterns in Motion

- Motion that is too fast or at a too fast frame rate can cause a correspondence problem.

  - **Correspondence problem**: brain must match points in in the left and right visual fields

Wagon-wheel effect:

https://www.youtube.com/watch?v=9MN5MF72PHs

- This is due to **temporal aliasing**:

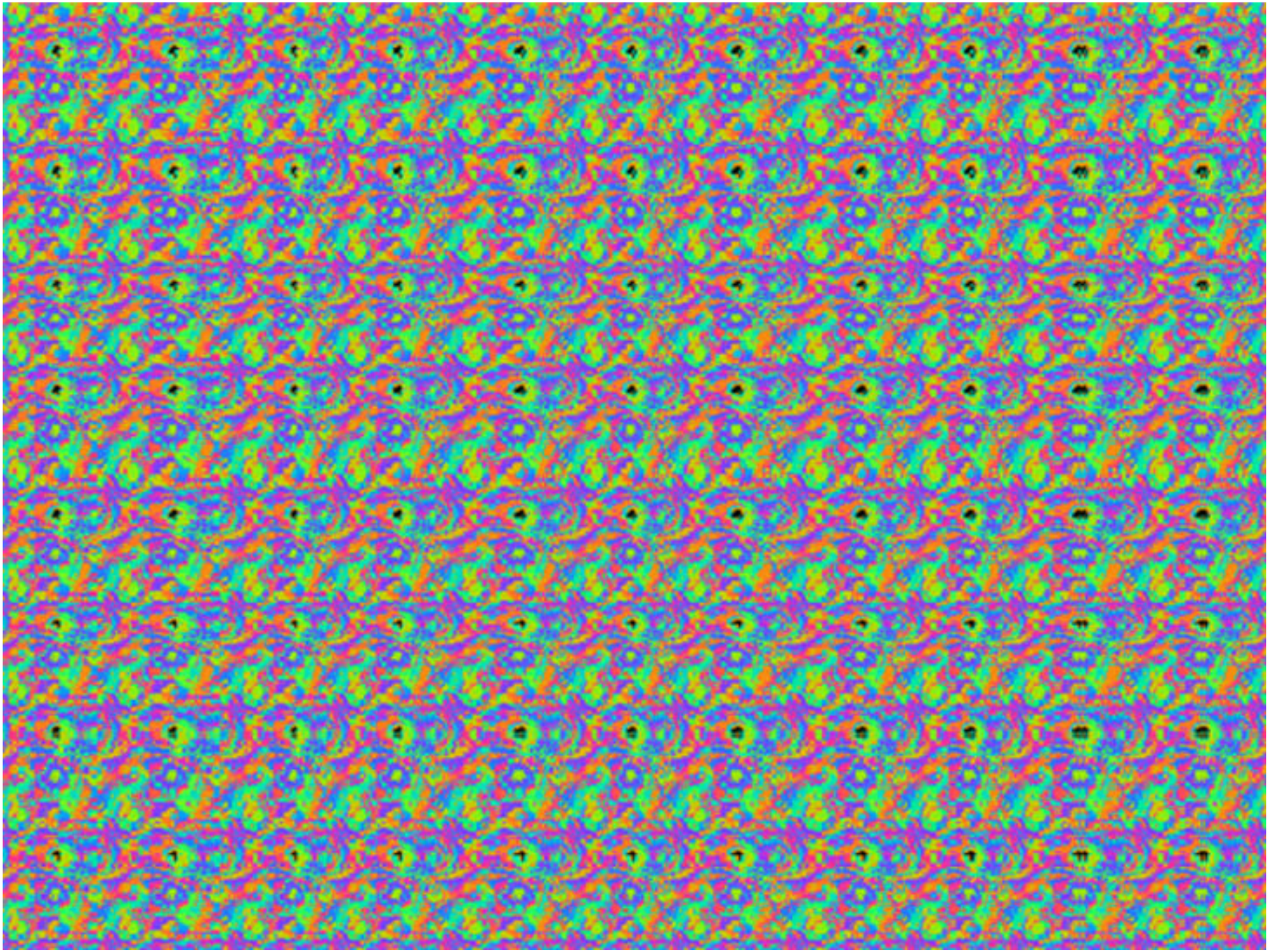
https://www.youtube.com/watch?v=ByTsISFXUoY

https://www.youtube.com/watch?v=x08vqAyfyrA

Correspondence problem

- If frame rate is adequate, movement is fast enough to fuse motion and also difference in **two positions is small enough** that the brain is able to ‘track’ the movement of an object.

- If this distance is too great for the time difference: the brain will confuse elements with the position of other elements, or the direction.



Motion in Graphics

- We can group things very quickly based on differences in how those objects are moving.

- We can treat relative motion as an attribute of a visual object, just like color or shape!

- How much should you move an object?

  - *Too fast* and you will have a correspondence problem.

  - *Too slow* and it will appear choppy and discontinuous (unpleasant)

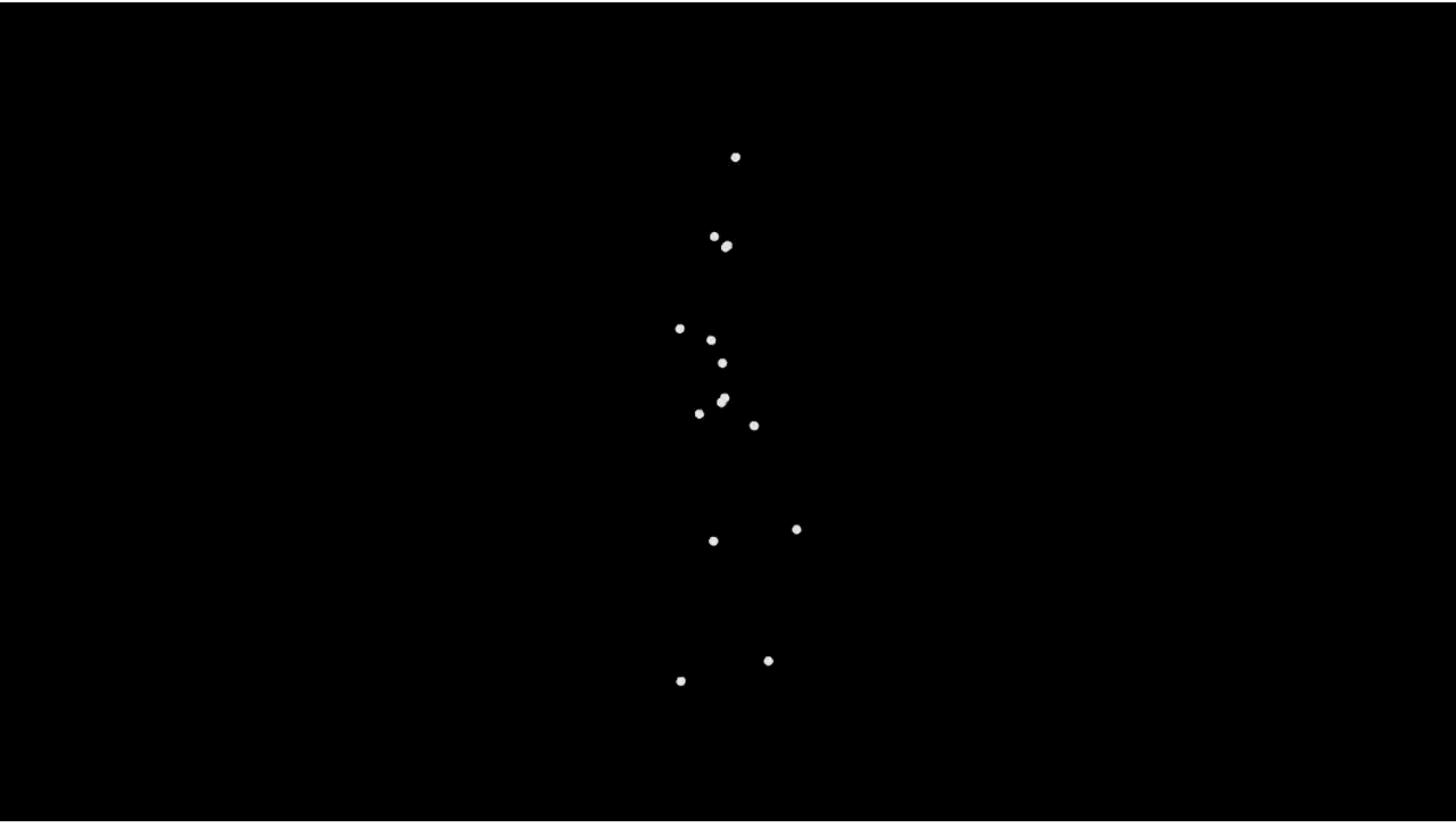
  - **Aim for 0.5 to 4 degrees per second of visual angle (about 0.5 to 4 cm/s at normal viewing distances for a typical screen)**

Perceptions of Causality

- If one thing appears to strike another, we perceive that the first object's movement *causes* the movement of the second.
  - This can be accomplished with relative motion, but timing is exquisitely sensitive to this perception!
    - Launching: w/in 70 ms of first object encountering second
    - Anything longer will have a qualitatively different relationship (e.g. entrainment, delayed launch, triggering)

- **Motion of biological origin**: our visual systems are especially sensitive to this motion.
  - You can reconstruct and recognize the motion of unconnected dots as the motion of a person with no other information.
  - Good for busting camouflage!

Sensitivity of Biological Motion

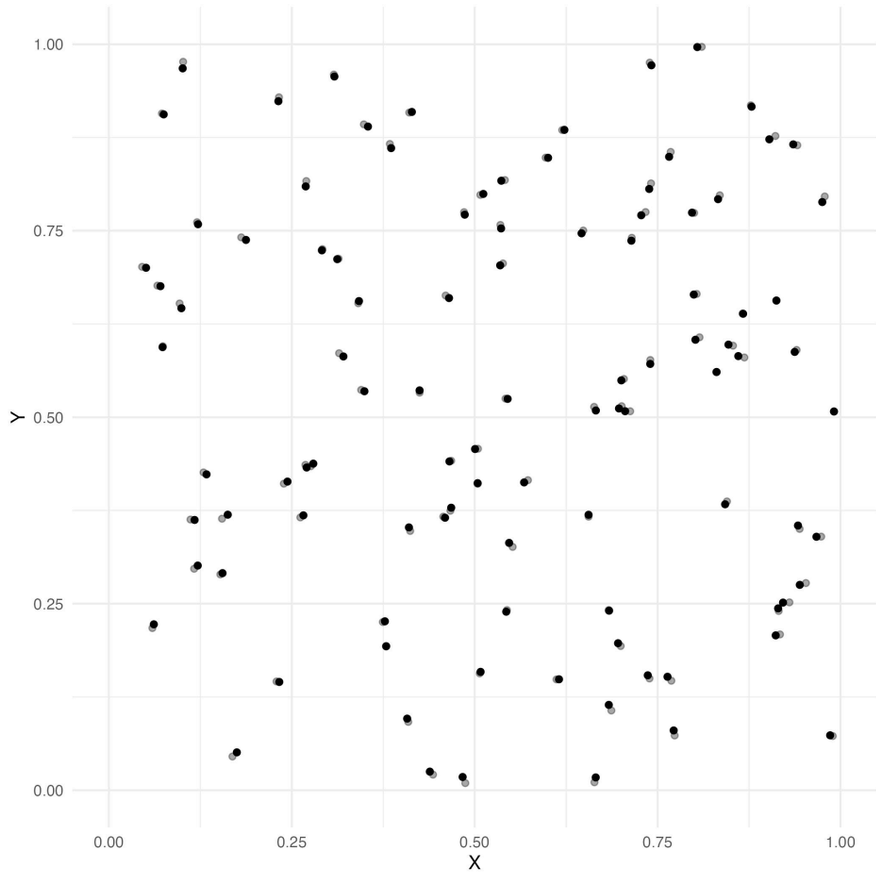


Creating Motion with R

- It's easy to create sequences of individual images in R by using loops and apply functions, they require stitching together in order to encode a video.
  - ImageJ is a tool that can do this with image sequences.

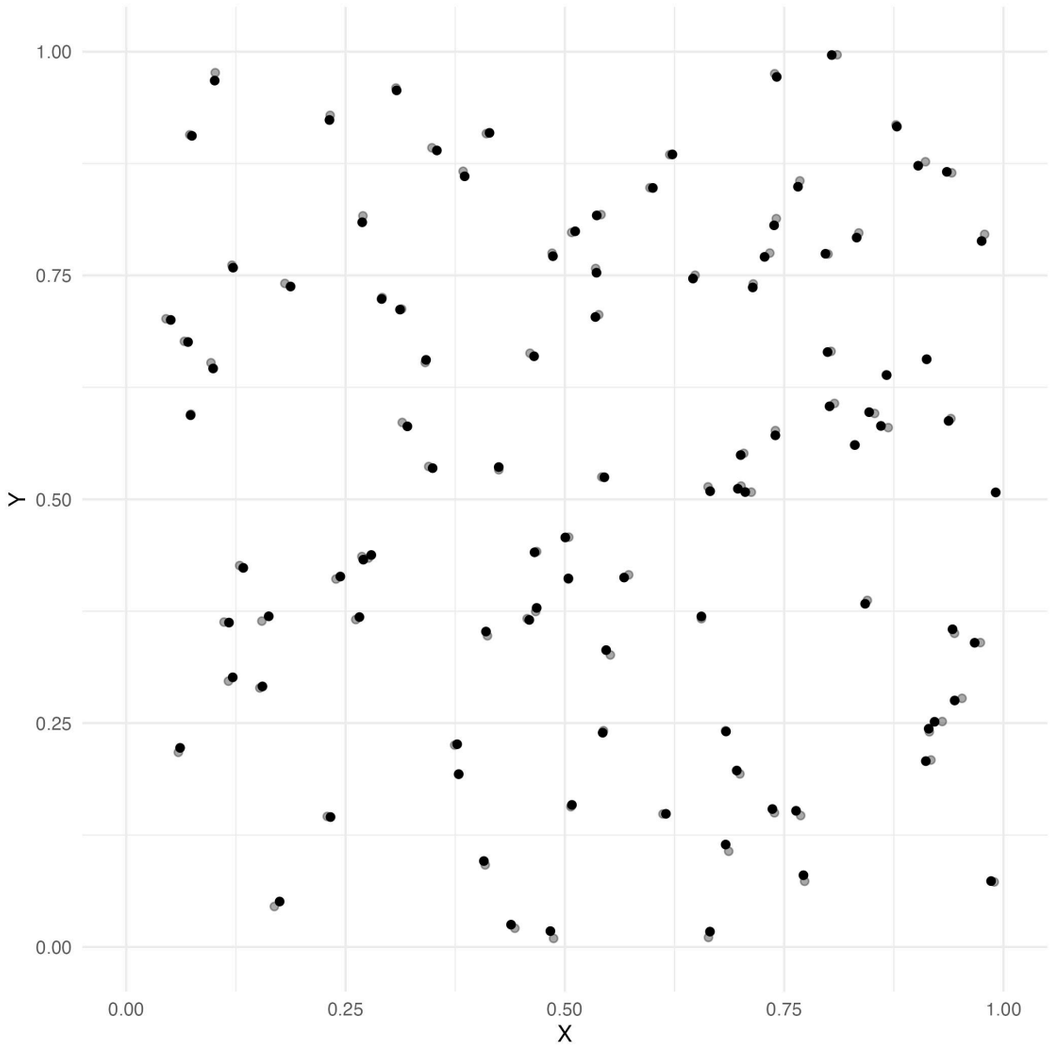
- Several packages can encode videos with R including av and gganimate.

- Two worries with encoding videos:
  - How will the individual images be compressed?
  - What codecs are available for playing?

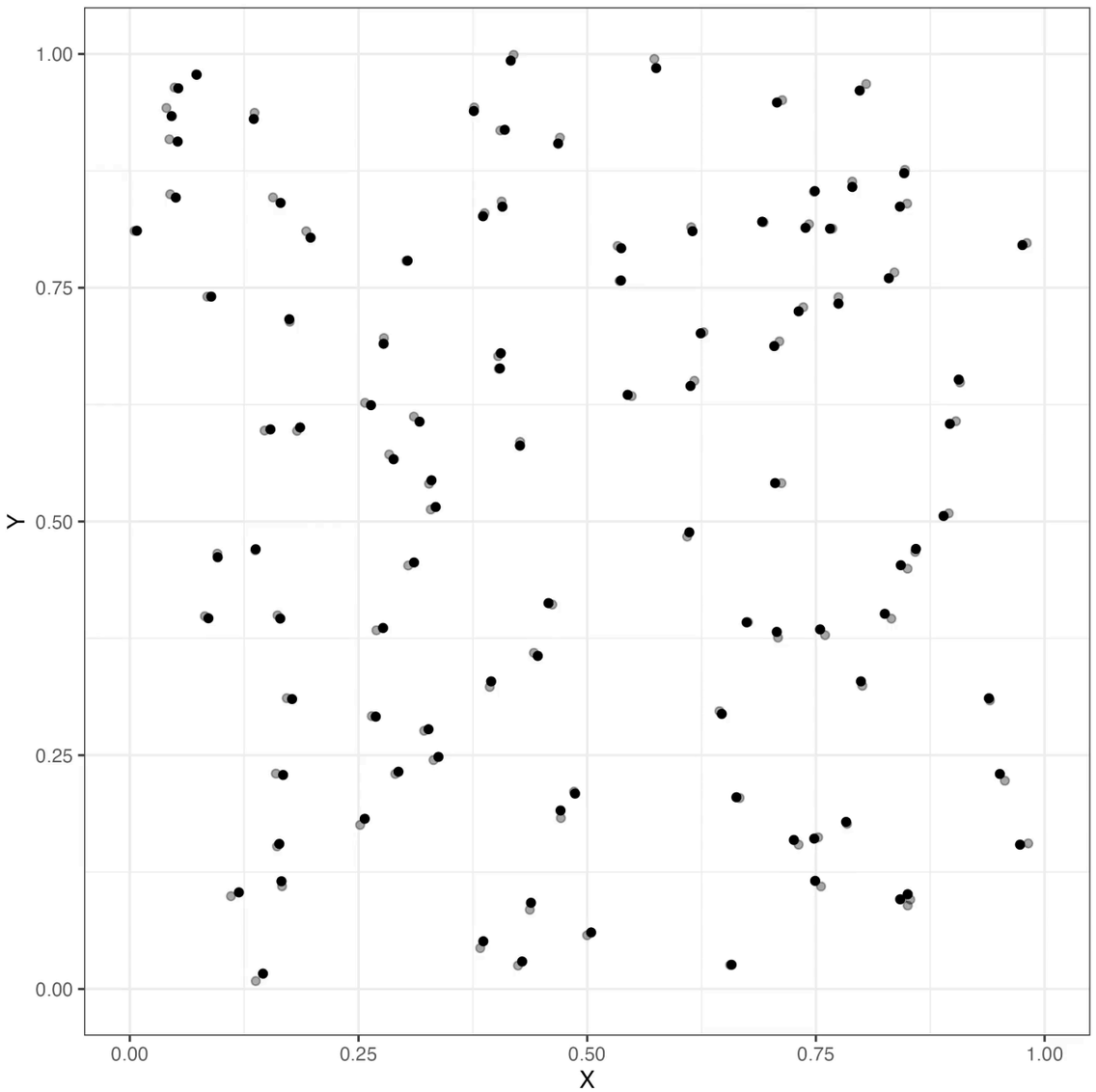


30 fps, Motion JPEG OpenDML, 14.2 MB
(Uncompressed size: 1.3 GB)

Creating Motion with R



ImageJ video
30 fps, Motion JPEG OpenDML,
14.2 MB
(Uncompressed size: 1.3 GB)



av video
30 fps, H.264, 537 KB

Group work:

Use the sample code to produce a visualization where movement is one visual attribute.

Use any data set to create any visualization.

Stitching together movies with ImageJ

1. Download ImageJ (if you haven't already): https://imagej.nih.gov/ij/download.html
2. Open ImageJ.
3. Under "File" select "Import > Image Sequence…" and then select the directory that contains your movie frames. NOTES:
   3.1. Your movie images MUST be in a directory alone. Any other files in that directory will cause the movie to fail!
   3.2. If you have a large movie, I suggest using a virtual stack.
4. Review your movie.
5. If you like it, then save it by selecting "File" then "Save As…" then "AVI". Select the frame rate and compression (JPEG works well). And then WAIT for it to finish saving before opening.