

Lookit coding exercise

To give video coding a try, you'll be coding a **single video** from a pilot of the intuitive physics study. (We're looking to both get a feel for your ability to follow the directions, and to give you a chance to check this is something you'd be up for doing over the summer!)

You can find the video at: www.mit.edu/~kimscott/urops/physics_example.mp4

You'll need the free software VCode (for Mac), which you can get here:

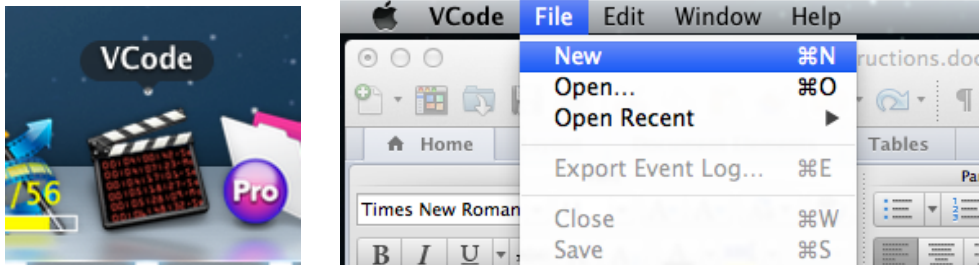
<http://social.cs.uiuc.edu/projects/vcode.html>

You're welcome to come in and do video coding in the lab (46-4013) if that's easier than finding a computer on campus to use – just let Kim (kimscott@mit.edu) know when you're coming by!

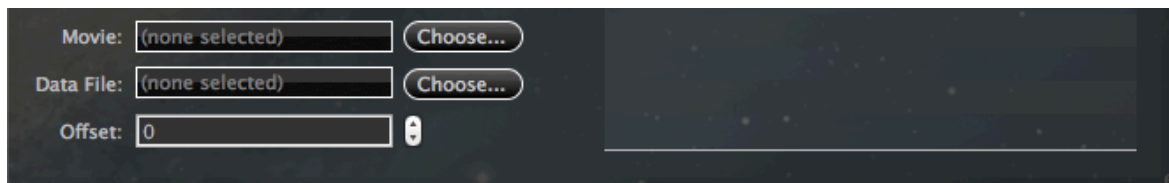
Following is a pared-down version of the “coding manual” covering just how to do preferential looking coding in VCode. For this coding exercise, please code **only the first 6 trials of the video**. When you're done, email the .cod and .txt files to kimscott@mit.edu. Please include a rough estimate of how long this exercise took you – we ask ONLY to calibrate the exercise for the future and make sure we're not imposing an unreasonable burden on UROP applicants.

Coding in VCode

You'll code each session video .mp4 using VCode. Open VCode and go to File→ New. (VCode doesn't open a window when you start.)



Open a video to code by clicking 'Choose...' next to 'Movie' (below, in the Admin panel).



Don't check 'interval playback mode' (bottom right corner of the movie window), or you won't be able to proceed frame-by-frame using the arrow keys.

Preferential looking coding

Create three event types called left, right, and away. Don't check the "range" box for any of them (this turns the events into "range events" which have a start and stop time, instead of point events which just occur at a particular time).

Key	Name	Color	Range
z	left	Red	<input type="checkbox"/>
x	right	Red	<input type="checkbox"/>
m	away	Yellow	<input type="checkbox"/>

The names matter (call them 'left', 'right', and 'away'), but the keys are up to you. Record the frames at which the child starts looking to YOUR left (by pressing 'z' in the example above), the frames at which she starts looking to your right ('x'), and the frames at which she starts looking away from the screen ('m'). After initially placing these marks, edit them to the nearest frame by

going back and dragging the edges. Of course you'll never be absolutely certain, but trust yourself a bit here—humans are very good at figuring out direction of gaze.

Trial markings

Add another non-range event called 'trial' and mark the first frame of each 'trial' of the video. Each video is actually many short video clips, concatenated together; they're labeled in red text at the top left. When the text changes (e.g. from 7-pref-phys-videos... to 8-pref-phys-videos...), that's a new trial. Include a trial marker at the very start of the file and at the very end. This is important to double-check VCode's idea of exactly how long the file is with ffmpeg's measurements of the duration of each component clip.

Behavior coding

Also add range events called 'peek', 'talk', 'point,' and 'fuss', and a non-range event called 'distract'.

peek	Any periods when the parent's eyes are open and facing the screen. Do mark even brief peeking.
talk	Any periods when the parent talks. Add a note to the event with what was said (you can use <code>"*inaudible*"</code> for inaudible speech or <code>"*noaudio*"</code> for speech where the video doesn't have audio). Don't worry about exact transcription if you can't hear well. The transcript is used to determine whether there was an obvious attempt to get the child to look/look away.
fuss	Any periods where the child is fussy (making fussy noises, squirming in an attempt to get off the parent's lap, or crying.) Add a note with a "level" of fussiness – 1 = fussy noises, trying to get off lap but doesn't seem too upset about it, 2 = unhappily trying to get off lap or actually crying. (And let me know if this feels like a weird system—we're at the very start of adding this so if you have a more natural categorization we can try that instead.)
point	Any periods where the parent points to the screen. If the parent is also peeking, add a note that says either 'left' (pointing to your left of screen), 'right', or 'general' (pointing at screen but not to particular event).
distract	Add when something external to the child happens that clearly explains a child's look away from the screen. Only mark if the child actually looks away at it. Add a note with what the distraction was—doesn't need to be in any particular format, these are for a human to read ☺ Distractions: parent talks, phone rings, loud noise outside, child drops toy Not distractions: child turns around to grab mother's glasses (the mother/glasses didn't do anything to precipitate this), child looks down at toy in hands.

Don't worry about coding any of these to the nearest frame: the timing information is so we can tell approximate duration and things that happen before lookaway vs. after.

Deleting marks

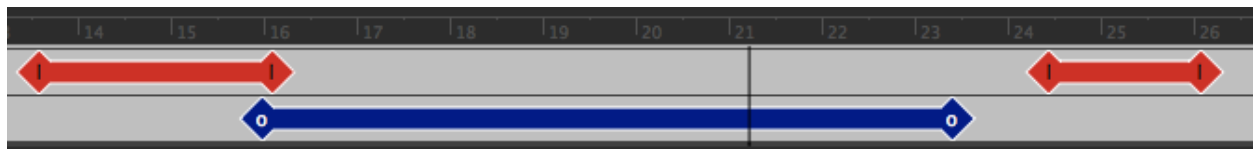
To delete an event, hold 'Alt' and click on it.

Invisible children

If you can't tell where the child is looking because his eyes are out of frame, use a range event called 'outofframe' and add an outofframe event for the duration of any periods where this applies. If you can see the side/back of the child's head and he's clearly not looking, or if he wandered into another room, etc. that's NOT outofframe. If you can see one of the child's eyes and can pretty much tell where/whether he's looking, also NOT outofframe.

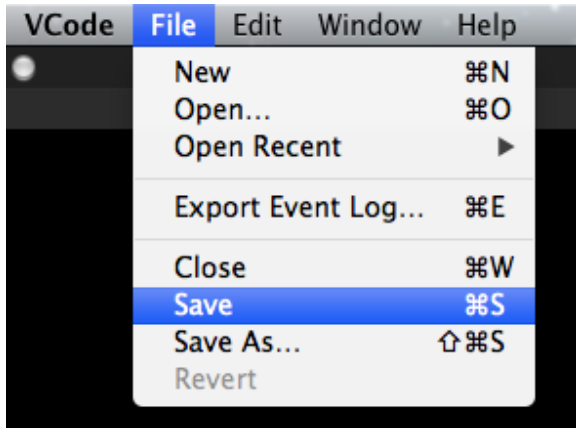
Outofframe takes precedence over any other coding during this period. Always code for looking time/preferential looking for the entire period not covered by 'outofframe' – otherwise, we may end up thinking (for instance) that the child looked away just before going out of frame. It's fine for looking and outofframe coding to overlap (if a frame is contained both in outofframe and in looking, we'll know the child is out of frame and give that priority).

Below is an example of outofframe coding (the lower blue bar) along with looking-time coding (the upper red bar). The child was looking at the screen when her eyes went out of view just before 16 seconds. When her eyes became visible again, she was not looking at the screen, but she looked again a bit past 24 seconds.



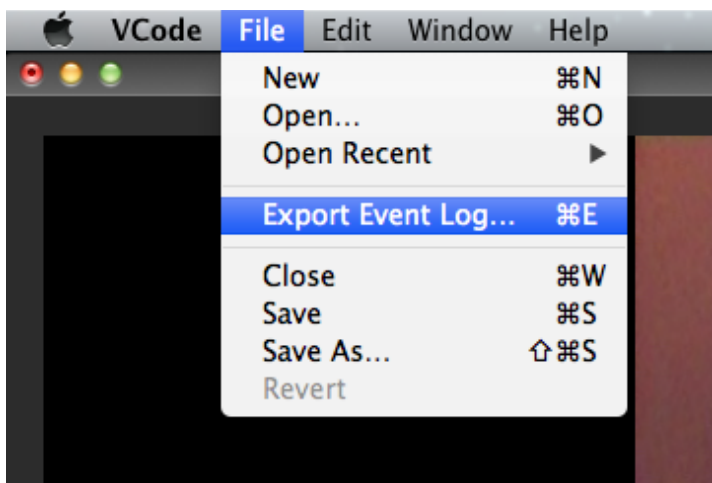
Saving your work

Once you're satisfied with the coded video, go to File -> Save:



Save the file as `physics_example_YOURNAME.cod`. This format makes it easy to go back and check or edit coding later, since we can easily see it overlaid on the video.

ALSO export the events you listed by going to File -> Export Event Log.



Save the text file as `physics_example_YOURNAME.txt`, which will be the default name.