VCode and VData

Tools for Video Annotation

Documentation by
Joshua Hailpern and Joey Hagedorn
University of Illinois at Urbana Champagin

VCode and VData

Tools for Video Annotation

VCode and VData are a suite of "open source" applications which create a set of effective interfaces supporting the video annotation workflow . Our system has three main components: VCode (annotation), VCode Admin Window (configuration) and VData (examination of data, coder agreement and training). The Design of VCode and VData was grounded in existing literature, interviews with experienced coders, and ongoing discussions with researchers in multiple disciplines. This document provides a description of the interface of VCode and VData, as well as documenation on set-up, and execution.

We greatly appriciate you using our system, and would love to hear any comments you have! From all of us in the Social Research Group in the Department of Computer Science at the University of Illinois, welcome to VCode and VData!

Joshua Hailpern, Joey Hagedorn, and Karrie Karahalios

Contents

Chapter 1:
Installation & System Requirements

Features

Multi-Video Stream Support:
Quite often, there are multiple streams of video that coders must annotate. This can be different camera angels, or even screen capture video on a computer. VCode presents one main video at full size, and a dock with other streams playing in real time. When a docked stream is clicked on, it repositions itself into the main video window, while the video which was the previous focus, scales down to the dock, thus equating visual importance with relative size and visual weight.

Different Events for Different Needs:
When annotating a video, we often have to capture different types of data. VCode supports multiple types of event annotation: ranged, momentary and notes/phonetic transcription. A ranged event is one which extends over a period of time (marking action start and duration). Momentary marks have no duration, and thus represent one specific moment in time. Comments can be attached to any mark, allowing additional observations, levels/ranking, or phonetic transcription (through onscreen phonetic keyboard). Any mark with a comment has a inverted outlines to signify that it has a comment attached.

Timeline & Secondary Data
The heart of VCode is the timeline. Events are graphically represented here by diamonds. The spatial-linear presentation allows users to not only see chronologically the location of their events, but also the relative position of one mark to another. Each "track" or "dependent variable" is a different color, allowing for quick and easy assessment of the annotations made to date. To avoid scrolling and maximize the use of screen real-estate, momentary events overlap, thus optimizing screen usage, while providing enough area for track isolation.

In addition to the annotator's events, additional secondary data can be displayed on the timeline, for example a waveform of the audio from the video. If data (from a computer, or even another coder's annotations) is logged in a separate log file, this data can be displayed graphically as a bar, line or scatter plot. As a result, video annotators can use the best information available, when making their decisions when to code.

Features

Multiple Playback Modes
Continuous playback is not always the preferred method of analyzing a video. Often multiple modes of playback are utilized; continuous or standard playback, continuous interval playback (play for N seconds, then s top), and skip interval playback (jump N seconds, then stop). This allows the video to be divided in to smaller segments for annotation of events that are more difficult to pinpoint (i.e. when a smile starts or ends). Though conceptually simple, manipulations of video using a standard VCR is often described as annoying, and due to hand eye coordination, repeatability & reliability may suffer.

Seperate Admin Window + Templates
To ensure consistent configuration between coders and sessions, all administrative features are consolidated in a single window. The expected workflow is such that a researcher would setup a single coding document with all the variables to be used on all the videos. This template would then be duplicated (with media and log files inserted for each trial). By using this model, large video files only need to be on a hard drive in one location, rather than embedded in the VCode file. Through the Admin Window, the name, color and hot key of each tack can be set through this list presentation. Tracks can be enabled as ranged events through a check box in this interface. The Administration Window is also where a researcher specifies videos and data file to be coded, as well as secondary data for contextual annotation. These elements are specified and synchronized through a drag and drop interface, all of which is hidden from the coder to prevent configuration corruption.

Coder Agreement Support
Critical aspects of the video coding workflow (training, reliability, and accuracy) revolve around demonstrating agreement between coders. VData (Figure 3) is a separate executable application specifically targeted to aid researchers in training and agreement analysis of coded data produced in VCode. VData calculates user agreement simply by dragging and dropping in VCode files. Users can set the tolerance variable to accommodate for variability in the mark placement by the coders.

It is not uncommon for multiple tracks or variables to be measuring slight variations on a theme (e.g. smiling vs. large smile vs. grin), thus VData implements a track-merging feature which allows opportunities on two distinct tracks to be treated indistinguishably. For a holistic view, researchers can select tracks to be added into a total agreement calculation. In other words, if analysis determines that a single track is not reliable or it is determined that a given track will not be used in the future, it can be easily excluded from the total agreement calcuation.

Features

Coder Conflict Resolution Support
We have optimized coder training and reliability analysis by providing a graphical mechanism to directly compare annotations of two coders. VData can create a VCode session containing specific tracks of two individual coders for side-by-side comparison. The visual, side by side, representation of the data makes it easy to recognize systematic errors in context and detect differences between two coders markings. This reduces the time necessary to locate discrepancies and discuss the reasons why they might have occurred. It is necessary to keep records of these agreement analyses performed with VData by text export. Maintaining export at each stage of the process provides additional transparency and maintains traceability of results that come out of the system.

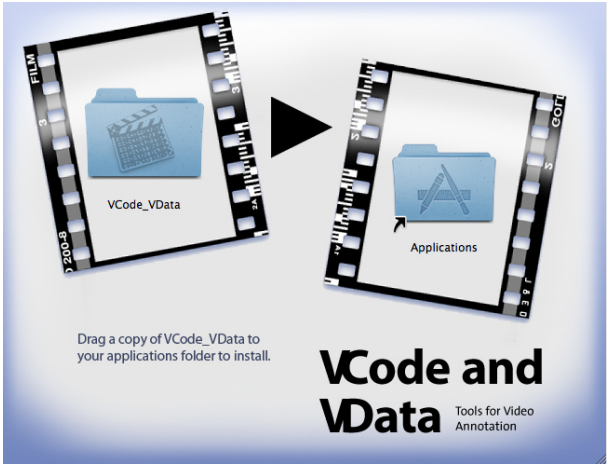
System Requirements

Mac OS 10.5.x (Leopard)
1.8 Ghz or Faster Processor (Intel or PPC)
Video in .mov format
1 Gig of Ram

Installation

1 Open the DMG file by double-clicking

2 Drag the folder on the left into the applications folder on the right.

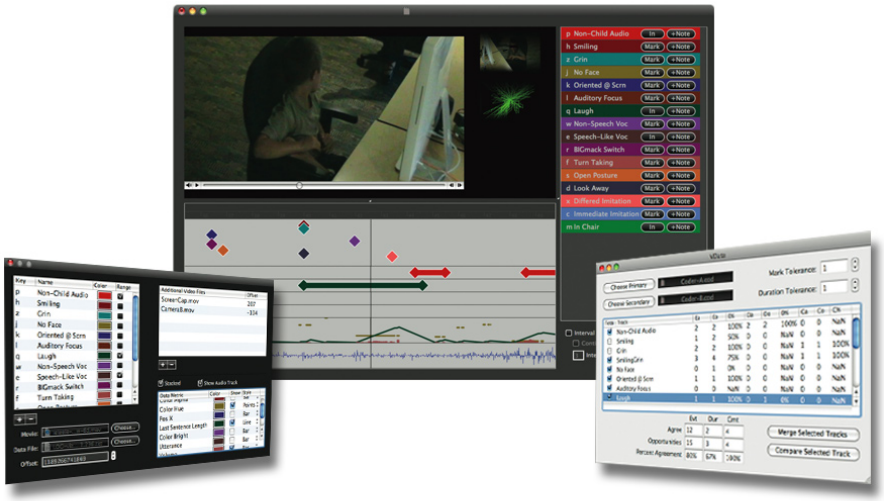


3 Run VCode and VData by clicking on their respective icons in your applications folder.

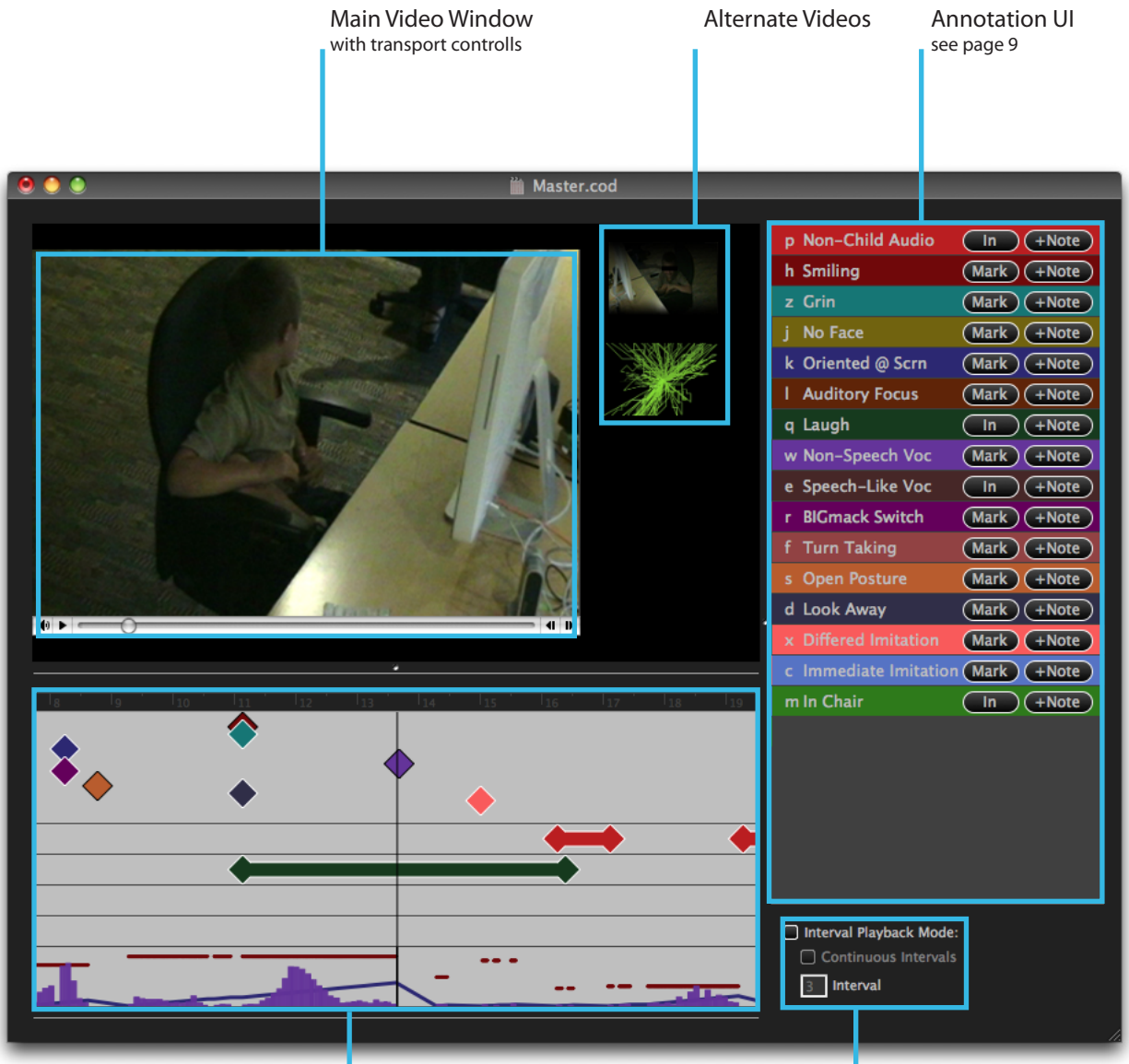
VCode

VData

Chapter 2:
Interfaces



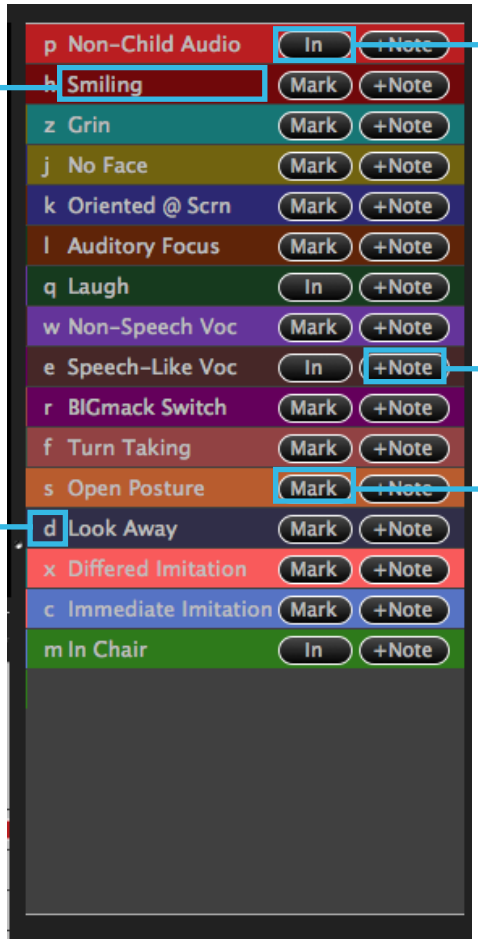
VCode



To facilitate multiple video streams VCode presents one main video at full size, and a dock with other streams playing in real time. When a docked stream is clicked on, it repositions itself into the main video window, while the video which was the previous focus, scales down to the dock, thus equating visual importance with relative size and visualweight.

Annotations can be inserted into the timeline via UI buttons or keyboard hot keys. To optimize the typically complex transport controls we isolated the key activities that coders need execute and provided controls limited to play/pause buttons, coarse and fine grained playhead positioning, and step controls.

VCode - Annotation UI



Annotation Name

Annotation Hot-Key
For annotation with keyboard interaction [as of version 1.2 vCode supports upper and lower case letters]

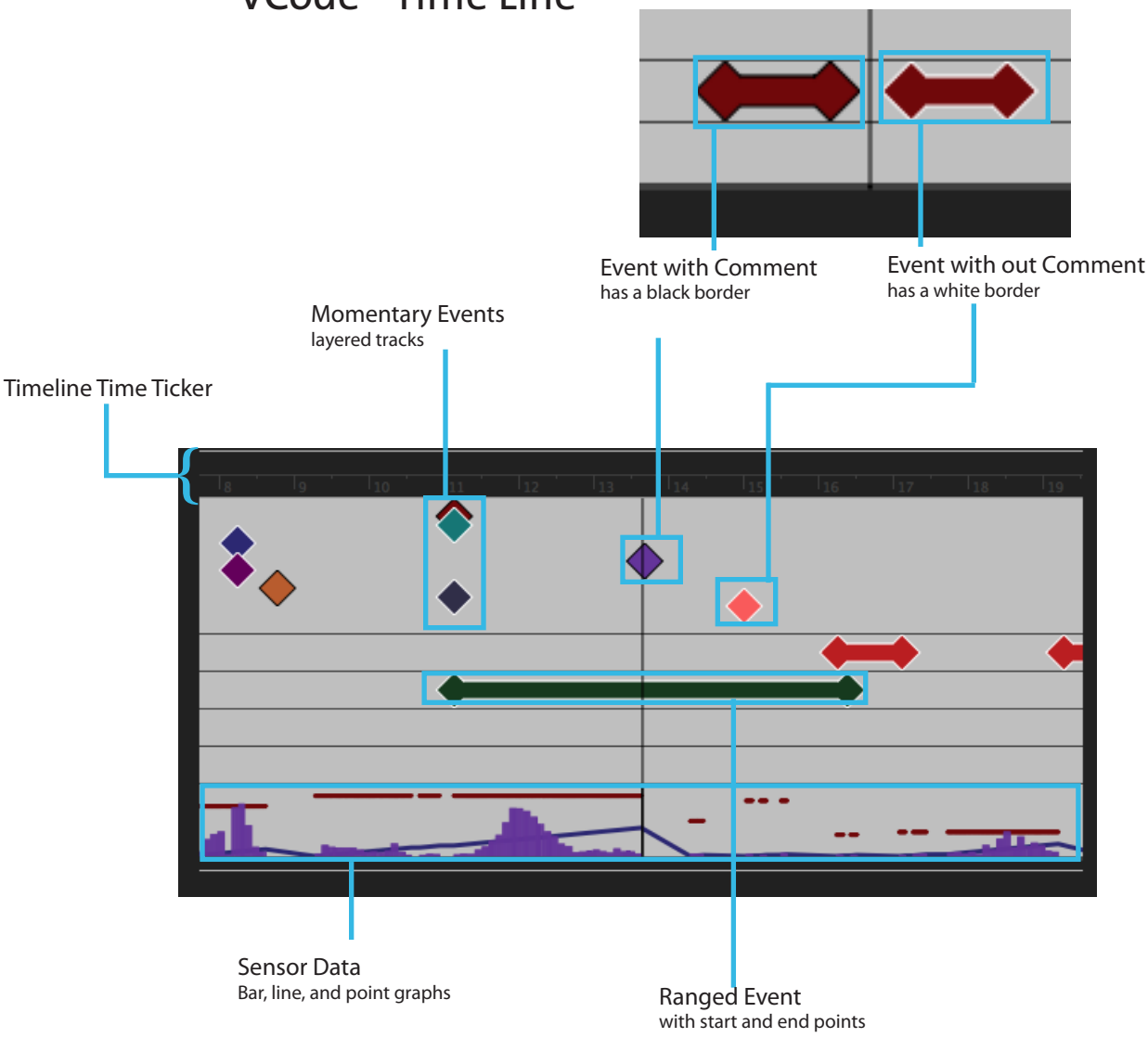
In/Out Button
For Ranged Events

Note
Apply a note to any annotation [hold down option+hotkey for keyboard interaction]
See page 11 for more.

Annotation Mark
For annotating momentary events

When annotating a video, two different classes of coding events emerge: ranged and momentary. A ranged event is one which extends over a period of time (marking action start and duration). Momentary marks have no duration, and thus represent one specific moment in time.

VCode - Time Line

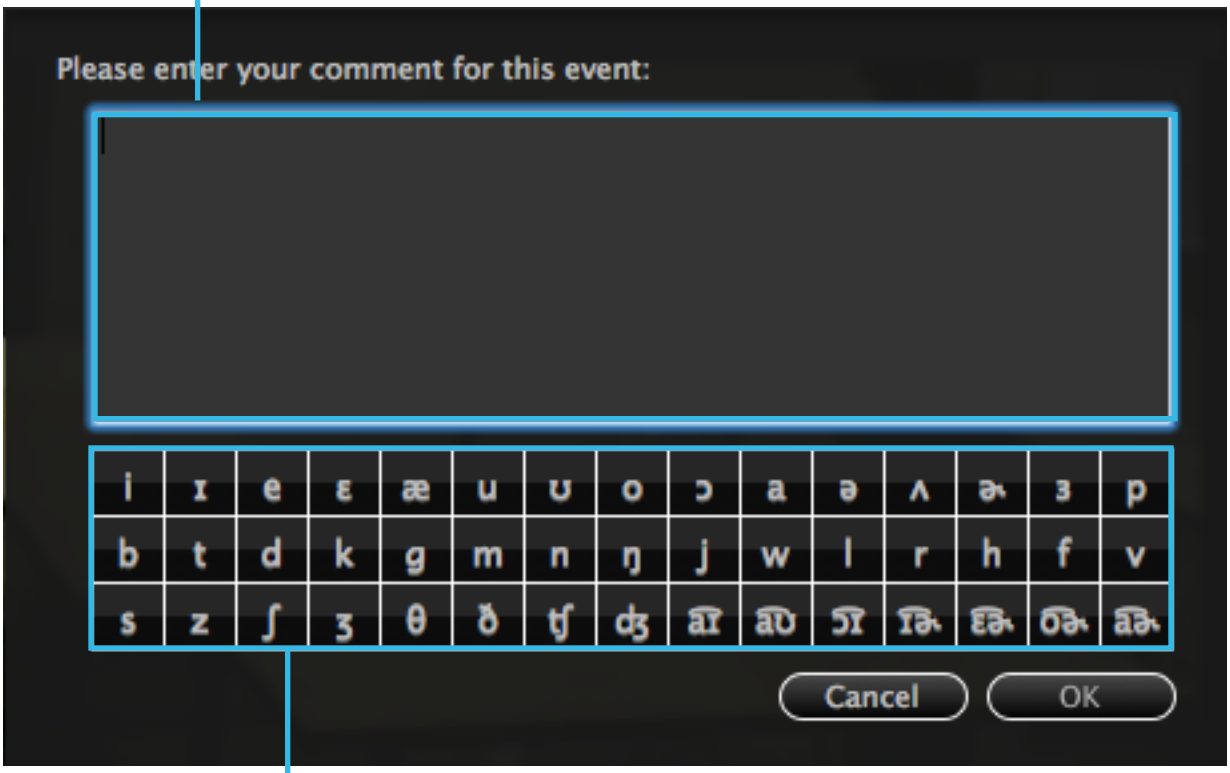


The heart of VCode is the timeline. Events are graphically represented here by diamonds. The spatial-linear presentation allows users to not only see chronologically the location of their events, but also the relative position of one mark to another. Each "track" or "dependent variable" is a different color, allowing for quick and easy assessment of the annotations made to date. To avoid scrolling and maximize the use of screen real-estate, momentary events overlap, thus optimizing screen usage, while providing enough area for track isolation.

In addition to the annotator's events, additional secondary data can be displayed on the timeline, for example a waveform of the audio from the video. If data (from a computer, or even another coder's annotations) is logged in a separate log file, this data can be displayed graphically as a bar, line or scatter plot. As a result, video annotators can use the best information available, when making their decisions when to code.

VCode - Comments

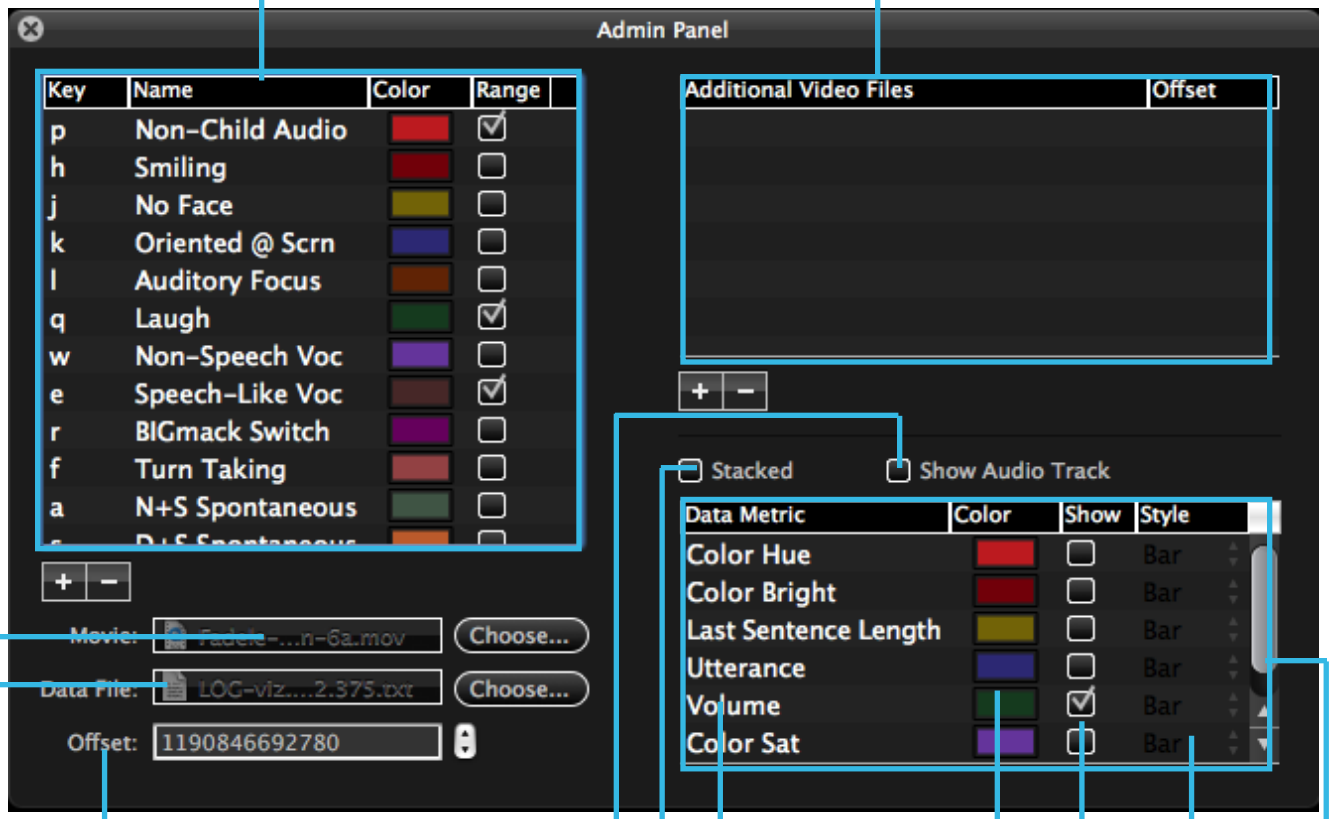
Comment Field
for comments, and/or rankings



Phonetic Alphabet
insets symbol into field

Comments can be attached to any mark, allowing additional observations, levels/ranking, or phonetic transcription (through onscreen phonetic keyboard). Any mark with a comment has a inverted outlines to signify that it has a comment attached.

VCode Admin Window



Track Configuration Field
ssee page 13

Additional Video Files
add multiple secondary video files, and set their "offset" so that all videos are alligned

Data/Video Offset
Sync the data file, and main video

Data File
The data file is specified format (page 16) with secondary data to show coders.

Main Movie
the main movie; where audio comes from. *(drag and drop)*

Stack Tracks
collapse all data tracks into only one line vs each track getting a line

Audio Track
display the audio from main movie as a wave-form

Data Metric Name
the name of the data metric

Color Config
what color to dis-play the data with

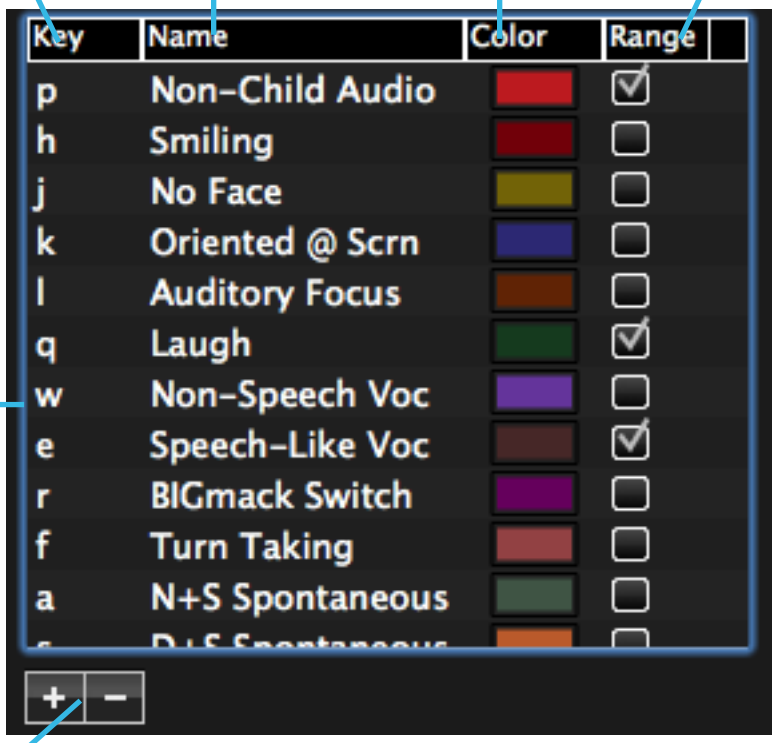
Show
whether or not to display said data

Presentation Style
set the presentation style of the data set (line, bar, dot)

External Data Config.
configures the presentation (which and how) of sensor or logged data to the user.

To ensure consistent configuration between coders and sessions, all administrative features are consolidated in a single window. The expected workflow is such that a researcher would setup a single coding document with all the variables to be used on all the videos. This template would then be duplicated (with media and log files inserted for each trial). The Administration Window is also where a researcher specifies videos and data file to be coded, as well as secondary data for contextual annotation.

VCode Admin Window



Hot-Key
Assign a hot key to a track to allow for easy keyboard marking

supports upper and lower case

Track Name
Assign a name to a track to allow easy identification by coders

Track Color
Assign a color to a track to allow easy identification by coders

Ranged Track
Check a box to set a given track to be a ranged event (start and end time) vs. a momentary event.

Adjust Track Order
Drag tracks up and down, to change their order and presentation.

Add/Remove Track
add a new track, or remove selected track

Researchers can add, remove, and reorder tracks which appear in a list format. The name, color and hot key of each tack can be set through this list presentation. Tracks can be enabled as ranged events through a check box in this interface.

VCode Keyboard Commands

Use Admin Assigned Hotkeys to mark tracks (both upper and lower case)

Option + hot key creates an annotation + comment

Option + click deletes a mark on the timeline

Left and Right arrow move forward one frame at a time. In interval playback mode, jumps between intervals.

Spacebar starts and stops playback

Ctrl+Left and Right arrow jumps between annotations

Apple+s saves

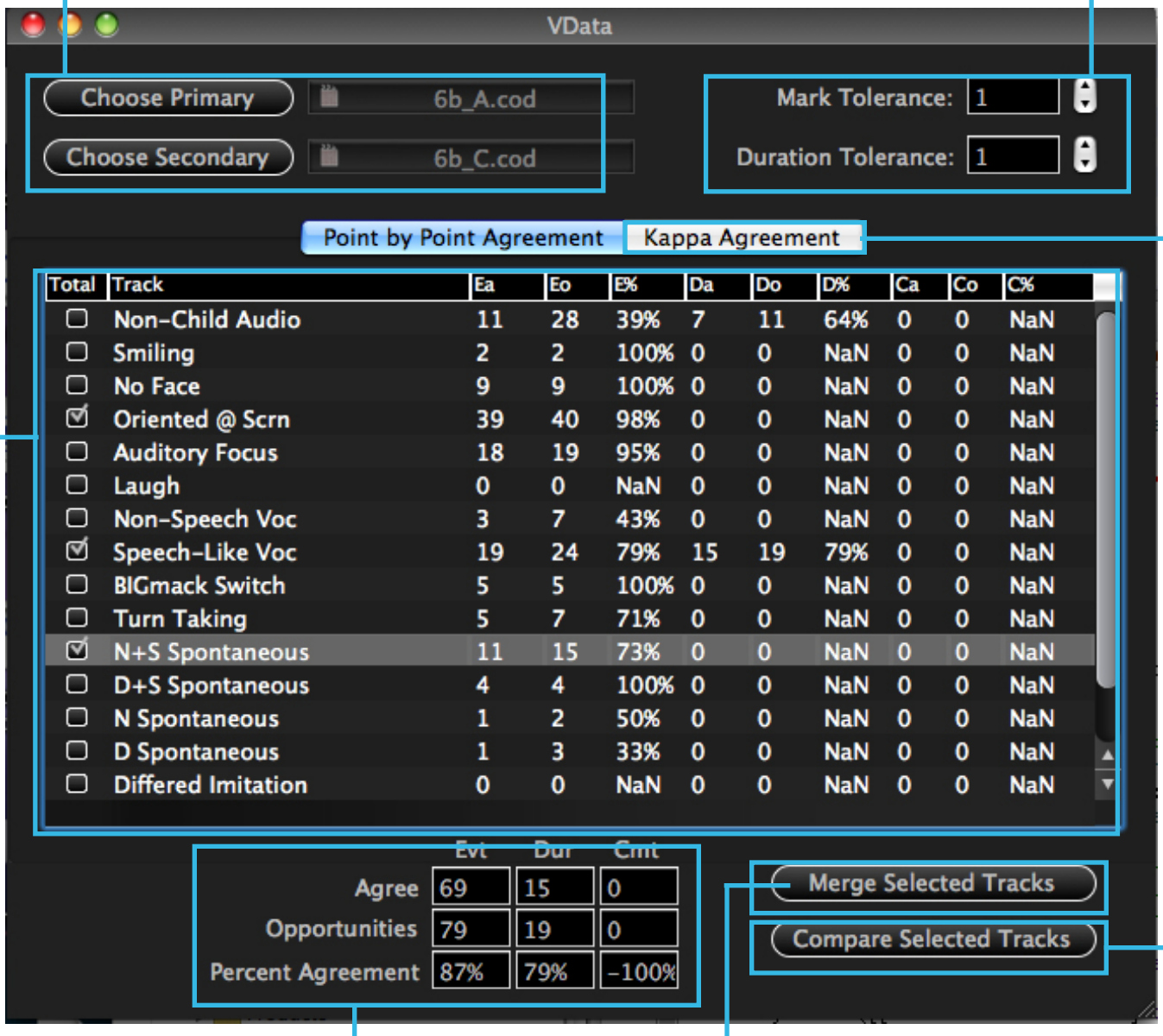
Apple+e exports

Kappa
Switch to Kappa mode. See page 17

VData

Set Primary & Secondary Coders
Through drag and drop, or file selection, choose the VCode file which is the primary coder, the VCode file to be the secondary coder.

Set Tolerance for Events
Set the "wiggle room" for matching events between coders; Mark Tolerance for momentary events (and start of ranged events) and duration for end point accuracy



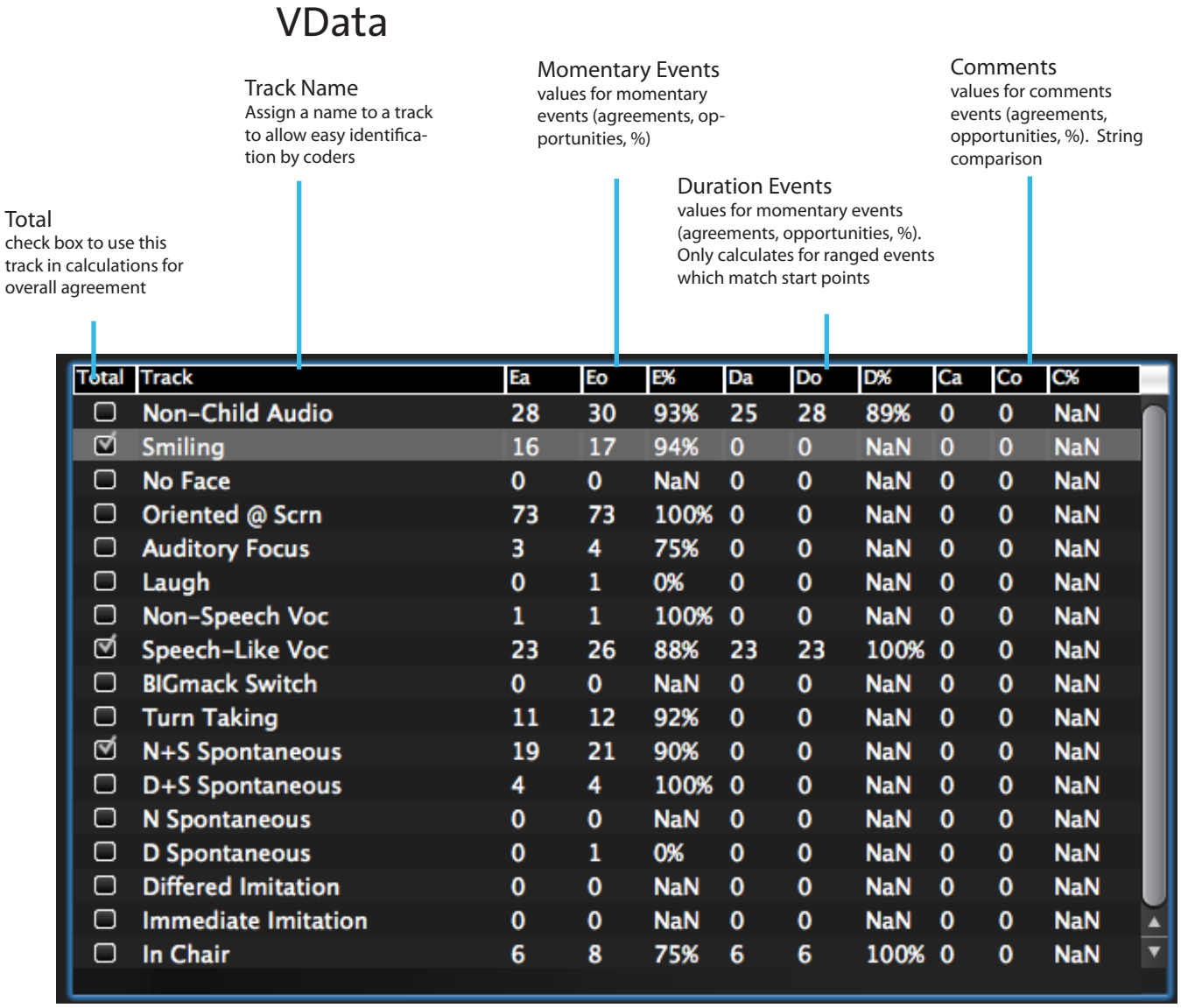
Track by Track Agreement
see page 16

Overall Agreement
Overall agreement calculations of selected tracks.

Merge Tracks
combine 2 or more tracks together and treat all points/coder as the same variable, for more corse grained analysis

Compare Tracks
see page 18

Critical aspects of the video coding workflow (training, reliability, and accuracy) revolve around demonstrating agreement between coders. VData is a separate executable application targeted to researchers in training and agreement analysis of coded data produced in VCode. See Page 15 for more details on calculations.



Total	Track	Ea	Eo	E%	Da	Do	D%	Ca	Co	C%
☐	Non-Child Audio	28	30	93%	25	28	89%	0	0	NaN
☑	Smiling	16	17	94%	0	0	NaN	0	0	NaN
☐	No Face	0	0	NaN	0	0	NaN	0	0	NaN
☐	Oriented @ Scrn	73	73	100%	0	0	NaN	0	0	NaN
☐	Auditory Focus	3	4	75%	0	0	NaN	0	0	NaN
☐	Laugh	0	1	0%	0	0	NaN	0	0	NaN
☐	Non-Speech Voc	1	1	100%	0	0	NaN	0	0	NaN
☑	Speech-Like Voc	23	26	88%	23	23	100%	0	0	NaN
☐	BIGmack Switch	0	0	NaN	0	0	NaN	0	0	NaN
☐	Turn Taking	11	12	92%	0	0	NaN	0	0	NaN
☑	N+S Spontaneous	19	21	90%	0	0	NaN	0	0	NaN
☐	D+S Spontaneous	4	4	100%	0	0	NaN	0	0	NaN
☐	N Spontaneous	0	0	NaN	0	0	NaN	0	0	NaN
☐	D Spontaneous	0	1	0%	0	0	NaN	0	0	NaN
☐	Differed Imitation	0	0	NaN	0	0	NaN	0	0	NaN
☐	Immediate Imitation	0	0	NaN	0	0	NaN	0	0	NaN
☐	In Chair	6	8	75%	6	6	100%	0	0	NaN

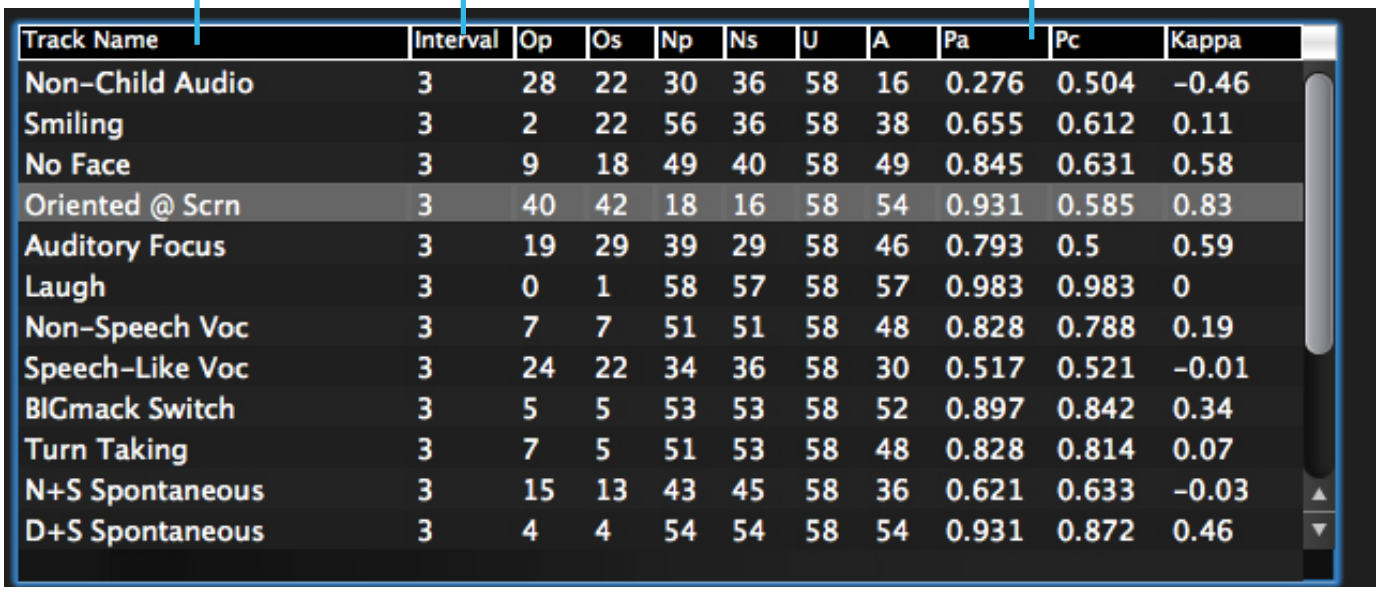
After files are specified, tracks are automatically loaded into the main data table which presents opportunities, agreements, and percentage agreement. VData calculates agreement for momentary events (E), Duration of matched events (D) and Comments (C). A more detailed description of Point by Point Agreement is located in Chapter 6.

VData

Intervals
Set the interval used for the specific track. Double click to change value. Default is last value used by Primary coder

Track Name
Assign a name to a track to allow easy identification by coders

Kappa Calculations
Please go to Chapter 6 for details on specific values here. Tooltips are available in program

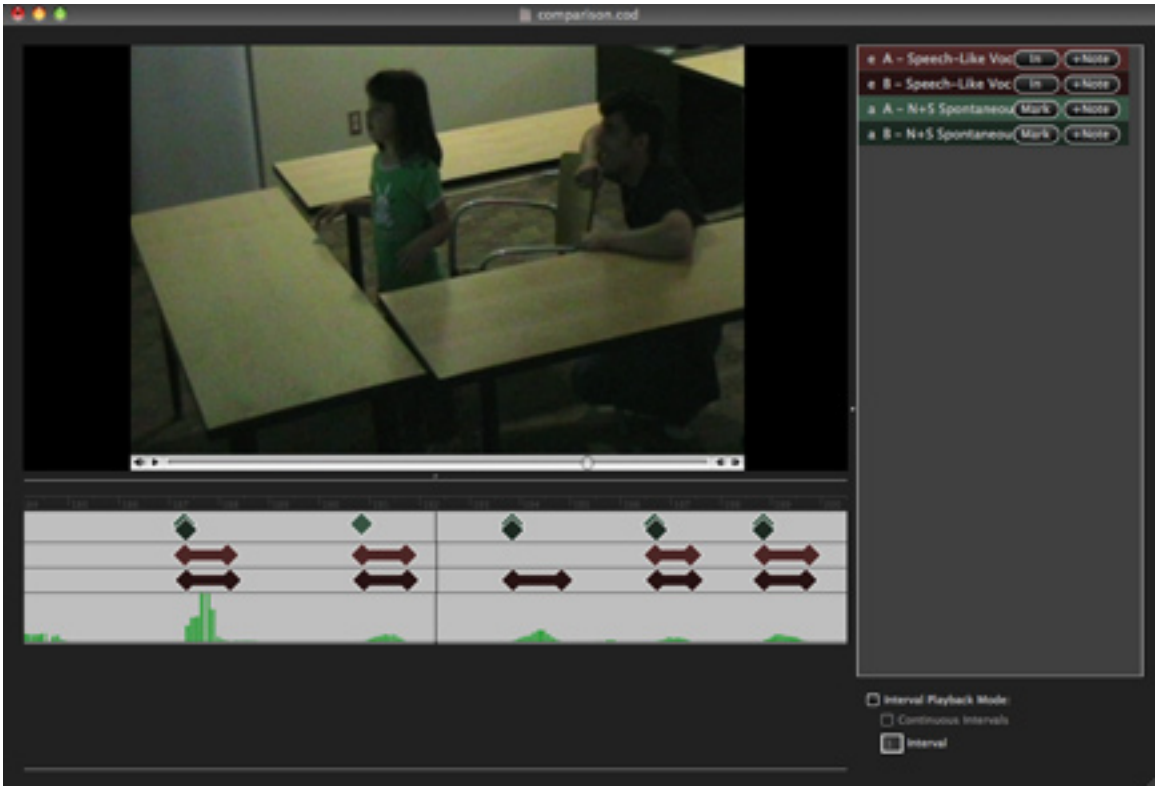


Track Name	Interval	Op	Os	Np	Ns	U	A	Pa	Pc	Kappa
Non-Child Audio	3	28	22	30	36	58	16	0.276	0.504	-0.46
Smiling	3	2	22	56	36	58	38	0.655	0.612	0.11
No Face	3	9	18	49	40	58	49	0.845	0.631	0.58
Oriented @ Scrn	3	40	42	18	16	58	54	0.931	0.585	0.83
Auditory Focus	3	19	29	39	29	58	46	0.793	0.5	0.59
Laugh	3	0	1	58	57	58	57	0.983	0.983	0
Non-Speech Voc	3	7	7	51	51	58	48	0.828	0.788	0.19
Speech-Like Voc	3	24	22	34	36	58	30	0.517	0.521	-0.01
BIGmack Switch	3	5	5	53	53	58	52	0.897	0.842	0.34
Turn Taking	3	7	5	51	53	58	48	0.828	0.814	0.07
N+S Spontaneous	3	15	13	43	45	58	36	0.621	0.633	-0.03
D+S Spontaneous	3	4	4	54	54	58	54	0.931	0.872	0.46

As of VCode 1.2, we introduced Kappa calculations. By definition, kappa can only be calculated for tracks with a set number of observation opportunities. As a result, calculations should only occur (be used) for tracks annotated in (Continuous) Interval playback mode. Other tracks will more than likely have strange numbers. See chapter 6 for detailed explanation of when to use kappa, and how to interpret results.

NOTE. Make sure video files are available to computer at original location to calculate kappa

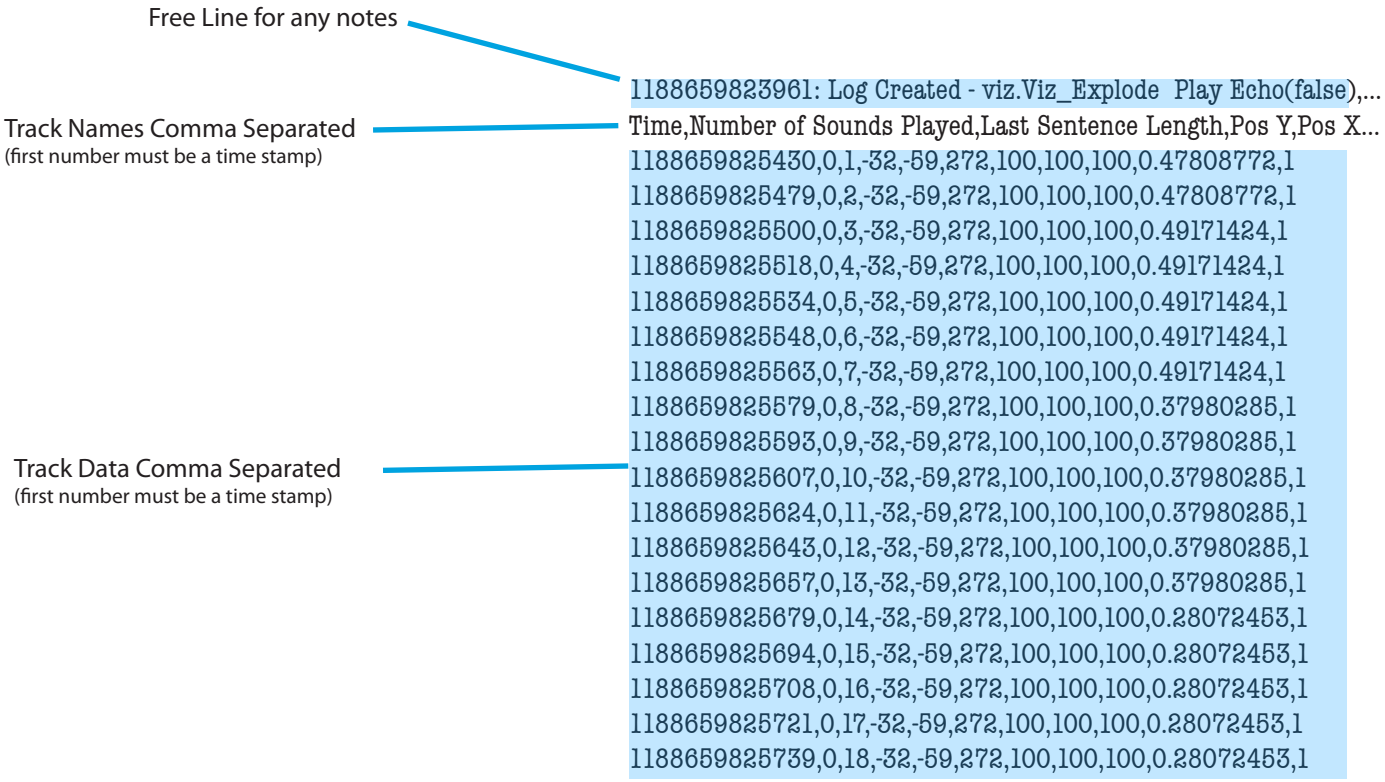
VData



We have optimized coder training and reliability analysis by providing a graphical mechanism to directly compare annotations of two coders. VData can create a VCode session containing specific tracks of two individual coders for side-by-side comparison. Each track selected for comparison shows up as a unique color, and it's shade is varied per coder. The visual, side by side, representation of the data makes it easy to recognize systematic errors in context and detect differences between two coders markings. This reduces the time necessary to locate discrepancies and discuss the reasons why they might have occurred.

Chapter 3:
Set -Up

Data Files


Free Line for any notes
Track Names Comma Separated
(first number must be a time stamp)
Track Data Comma Separated
(first number must be a time stamp)
1188659823961: Log Created - viz.Viz_Explode Play Echo(false),...
Time,Number of Sounds Played,Last Sentence Length,Pos Y,Pos X...
1188659825430,0,1,-32,-59,272,100,100,100,0.47808772,1
1188659825479,0,2,-32,-59,272,100,100,100,0.47808772,1
1188659825500,0,3,-32,-59,272,100,100,100,0.49171424,1
1188659825518,0,4,-32,-59,272,100,100,100,0.49171424,1
1188659825534,0,5,-32,-59,272,100,100,100,0.49171424,1
1188659825548,0,6,-32,-59,272,100,100,100,0.49171424,1
1188659825563,0,7,-32,-59,272,100,100,100,0.49171424,1
1188659825579,0,8,-32,-59,272,100,100,100,0.37980285,1
1188659825593,0,9,-32,-59,272,100,100,100,0.37980285,1
1188659825607,0,10,-32,-59,272,100,100,100,0.37980285,1
1188659825624,0,11,-32,-59,272,100,100,100,0.37980285,1
1188659825643,0,12,-32,-59,272,100,100,100,0.37980285,1
1188659825657,0,13,-32,-59,272,100,100,100,0.37980285,1
1188659825679,0,14,-32,-59,272,100,100,100,0.28072453,1
1188659825694,0,15,-32,-59,272,100,100,100,0.28072453,1
1188659825708,0,16,-32,-59,272,100,100,100,0.28072453,1
1188659825721,0,17,-32,-59,272,100,100,100,0.28072453,1
1188659825739,0,18,-32,-59,272,100,100,100,0.28072453,1

Video Files

Most Quicktime Video file formats are supported for VIDEO playback. Audio playback should equally be supported, though not all audio formats support the graphical representation of audio.

We personally, have tested with the audio codec in the file as: "QDesign Music 2, Stereo, 48.000 kHz", but we do think it should work with more standard formats such as AAC or Linear PCM. The source may need to be stereo (though we only visualize the left channel).

You should be able to change these settings when you export from Quicktime if you choose a custom video format.

Chapter 4:
Using VCode

VCode Workflow

The Ideal model for using VCode for Video Annotation in a group setting revolves around templates.

1. The researcher should set up a VCode file with all the specified tracks (variables) he/she plans to record. For each track, a color, and hot key should be specified. This template is referred to as a Generic Template (GT)

2. Replicate this GT for each video/trial/session you plan to annotate. We call these Trial Templates (TT).

    2.a. With each TT. choose the main video file
    2.b. Chose the data file (if present) and sync the data file with the main video file
    2.c. Add any additional video files to be used
    2.d. Sync the additional videos (if present) with the main video
    2.e. Specify any secondary data to be displayed (if present), and set the color and display style.

3. Have each coder duplicate the TT they need to use for their annotation, have have them append a Unique ID, leter or Number to the end of the file name.

4. Have the coder annotate the video, and enable/disable the playback mode accordingly (via the check boxes and time span).

Exporting
Raw data can be exported at any time, to a CSV text file. This can be used for record keeping, or for performing complex calculations across multiple data points. Each even occurs on a separate line, with the duration information (if present) and comments.

Chapter 5:
Using VData

VData Workflow

VData has many use models including reliability checks, training, and data analysis. Though Point by Point agreement calculations can be made without having the video or data files on your computer, Kappa does require them. If those files are not present, VData will run as normal, but display "nan" for all kappa values. We describe the basic set up, and some more specific options of analysis and agreement checks possible:

1. Take two VCode files, with the same source material, and load them into VData. This can be done either by dragging and dropping, or by using the buttons to open a file dialogue. If you want to perform kappa calculations, make sure you have your video files.

2. Examine your data, track by track.

Tolerance:
It is not uncommon for multiple tracks or variables to be measuring slight variations on a theme (e.g. smiling vs. large smile vs. grin), thus VData implements a track-merging feature which allows opportunities on two distinct tracks to be treated indistinguishably.

Total Agreement
For a holistic view, researchers can select tracks to be added into a total agreement calculation. In other words, if analysis determines that a single track is not reliable or it is determined that a given track will not be used in the future, it can be easily excluded from the total agreement calculation.

Kappa Agreement
For those tracks that were annotated using interval playback mode, you are able to switch TABS, and examine the kappa values. Feel free to adjust the interval to match the intervals used for each track. We present all the sub data/calculations so that you can examine the raw data used to calculate your kappa value.

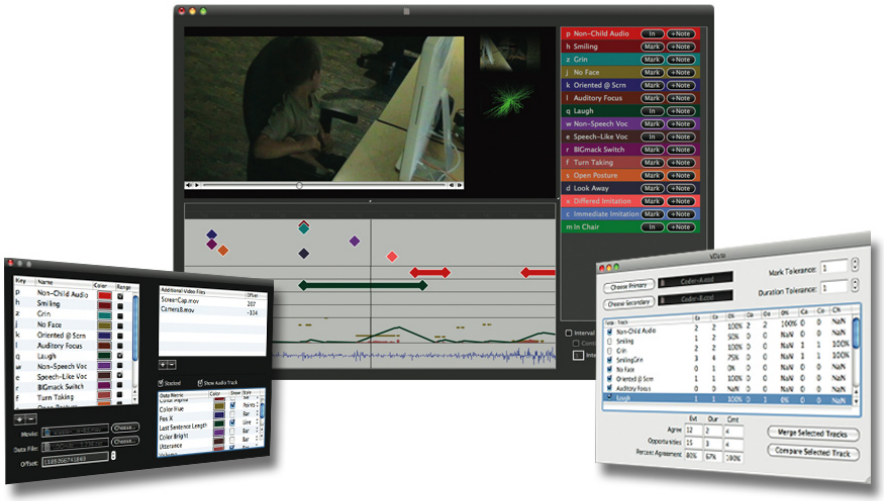
Conflict Resolution
We have optimized coder training and reliability analysis by providing a graphical mechanism to directly compare annota-

VData Workflow Cont

tions of two coders. VData can create a VCode session containing specific tracks of two individual coders for side-by-side comparison. The visual, side by side, representation of the data makes it easy to recognize systematic errors in context and detect differences between two coders markings. This reduces the time necessary to locate discrepancies and discuss the reasons why they might have occurred. It is necessary to keep records

Exporting
Reliability data can be exported at any time, to a CSV text file. This can be used for record keeping, or for performing more complex calculations across multiple data points. This includes both Point by Point and Kappa data. If VData could not locate the video files, necessary for kappa, nan will appear as the values when exported.

Chapter 6:
Understanding Agreement



Point by Point Agreement

Brief Summary of Point by Point Agreement

Point by Point agreement allows for an examination of events that can occur at any point on an infinite time spectrum (such as that in a video file). The agreement relies on two coders, one of who is the Primary coder, the other, Secondary. For all calculations in a given experiment the same coder should be the Primary coder.

Point by Point agreement breaks annotations up into Opportunities and Agreements. For each event an opportunity is said to occur when the primary coder makes a mark. If the secondary coder also makes a mark, within a tolerance, the marks are said to agree. Point by Point agreement is calculated as a raw percentage.

Calculation of Point by Point Agreement

O = Opportunities for agreement
A= Agreements between coders

Point by Point Agreement = 100 * (A/O)

Making Sense of Your Number

Kazdin (see reference at bottom of page) uses 80% as a benchmark for good agreement. However, this number can be lowered under certain circumstances. Specifically, if once can factor in the "difference" from error, and still see a significant difference in the raw data.

Kazdin, A. E. Single-Case Research Designs: Methods for Clinical and Applied Setting. Oxford University Press, USA, 1982.

Kappa

Brief Summary of Kappa
Kappa calculations for agreement is based upon the notion that a given time span (the video duration) can be broken down into discrete segments called opportunities. By Opportunity we mean, that in a certain period, an observation can be made (aka something DID or DID NOT occur). The presence of a mark, means it occurred. The absence of a mark means it did NOT occur. Kappa examines the number of agreements between two coders for events occurring and NOT occurring.

NOTE: *Kapa should only be used for momentary events made in a interval playback mode.*

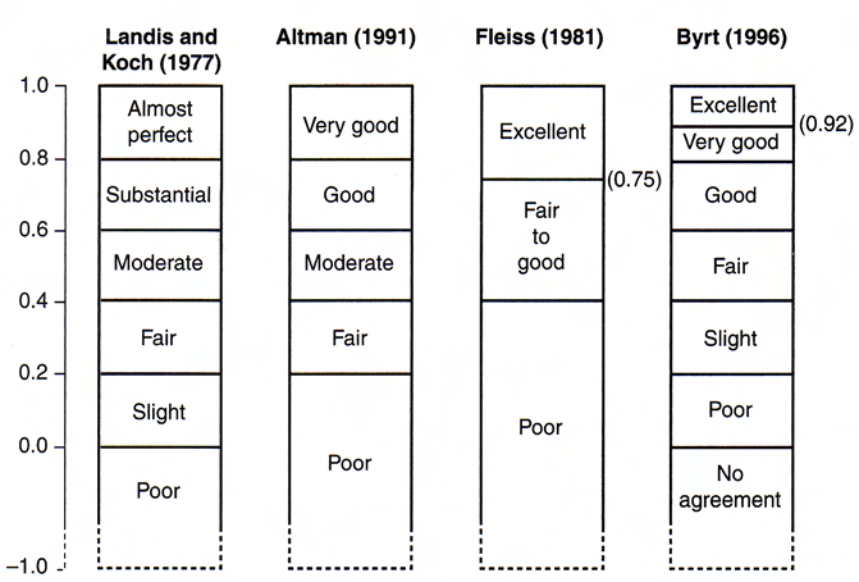
K=kappa
$\mathbf{O_X}$ = the number of occurrences of a mark for coder X
$\mathbf{N_X}$ =the number of nonoccurrences of a mark for coder X
U = number of opportunities for agreement
= video duration/interval playback time
A = number of agreements for both occurrences AND nonoccurrence between both coders *(how many places do coder 1 and 2 agree that an event did happen + the number of places coder 1 and 2 agree that an event did not happen).*

$\mathbf{P_a} = A / U$
$\mathbf{P_c} = ((O_1 * O_2) + (N_1 * N_2))/U^2$

$\mathbf{K = (P_a - P_c)/(1-P_c) = kappa}$

see next page for kappa interpretation

Kappa



Proposed classifications for the interpretation of a kappa value.

Kappa can be interpreted in a number of ways. The above chart shows 4 accepted ways of interpreting your Kappa values. The title for each bar is the authors and date of the publication from which theat scale comes from.

Graph taken from *Epidemiology: Beyond the Basics* by M. Szklo and F. Javier Nieto. Jones and Bartlett Publishers. 2006.