# Table of contents

# 5 Development of a new Lookit platform for collaborative use by multiple labs

Our eventual vision for the Lookit platform was never simply as a tool for our one lab to conduct specific research: its true potential lies in democratizing both participation and data collection and lowering the barriers to rigorous, replicable research for all investigators. Having demonstrated proof of concept for online developmental data collection using the initial Lookit prototype (Chapters 3 and 4), we next turned our focus to the development of a more robust platform that would simplify study creation and management and allow many researchers to run their own studies on Lookit. For this phase of the project we partnered with the Center for Open Science, an institution specifically dedicated to advancing openness and replicability of research, for software development services.

In this chapter, we first give an overview of functionality of the new Lookit platform and describe how a researcher specifies a study protocol. Next we turn to a series of five studies used for beta testing, each either currently collecting data or at least partly implemented. Finally we will discuss the current state of tools for processing and analyzing data collected on Lookit, the quality of data collected on the new site, and the challenges that remain. Note that while this chapter provides a broad overview of the functionality of the platform, it is not intended to replace ongoing documentation.

## 5.1 Overview of user interfaces and functionality

The platform is now split into two main components: Lookit (the participant-facing site, where families log in and can take part in studies) and Experimenter (a researcher-facing site, where researchers create, test, and manage studies and download data; this is an instance of a more general behavioral data collection tool developed by COS). Code for both pieces is available via github (Lookit: https://github.com/CenterForOpenScience/lookit-api; Experimenter: https://github.com/CenterForOpenScience/experimenter). Each piece also relies on specific experimental components defined in a separate repository called exp-addons (https://github.com/CenterForOpenScience/exp-addons) and on a player for the studies (https://github.com/CenterForOpenScience/ember-lookit-frameplayer).

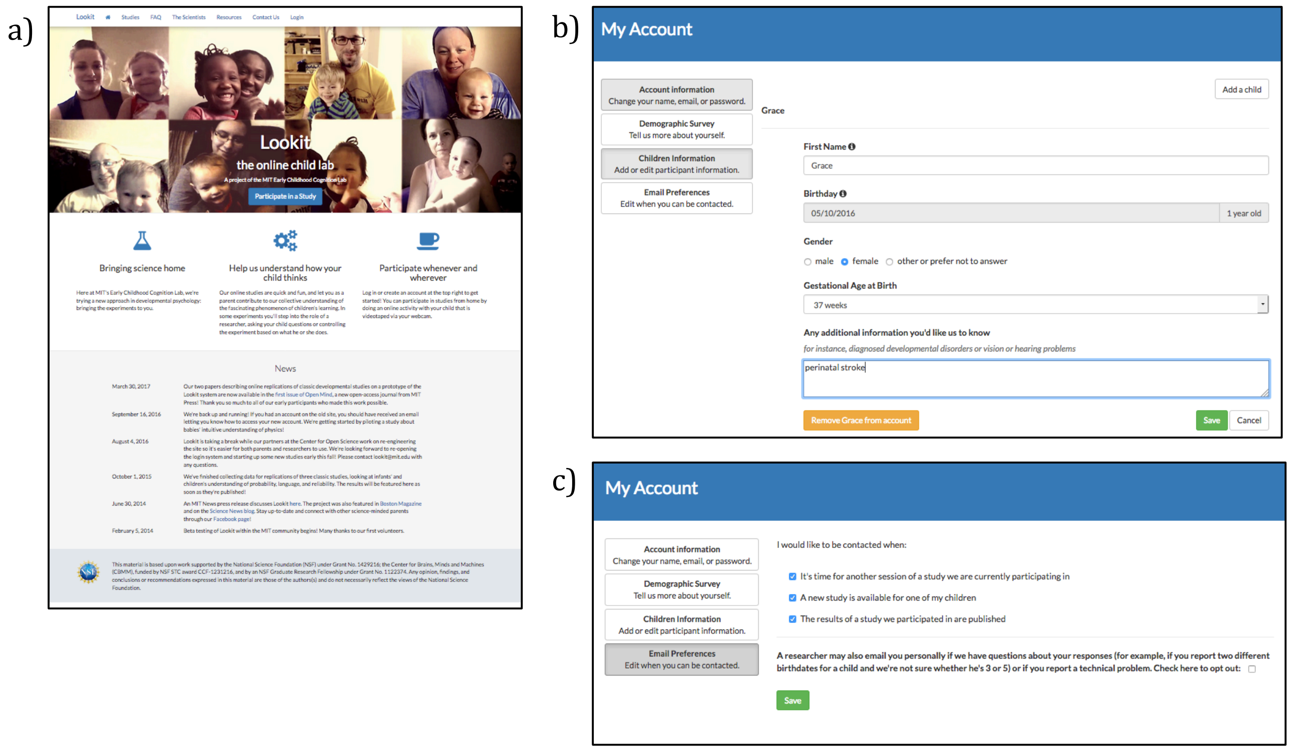



Figure 5.1. Sample screenshots from participant-facing Lookit site. a) Lookit home page. b) Registering a child or editing child information. c) Setting email contact preferences.

**Participant experience.** The Lookit website (https://lookit.mit.edu, see Figure 5.1) is publicly accessible and anyone can create an account. No software download is required. At registration, the parent provides a username, contact email address, and password. The contact email may be changed later. The parent can then register any number of children; for each child he or she provides a birthdate, nickname, gestational age at birth in weeks, and any additional comments (e.g. diagnosed developmental disorders).

Also under 'My account,' the parent can complete a general demographic survey (indicating family structure, educational attainment, number of children in the home, racial identification, geographic setting (city/state and urban/rural/suburban) and set email preferences.

The parent can select a study to participate in under 'Studies' (see Figure 5.2). Each study detail page displays the study name, description, and purpose; eligibility criteria; duration; and name and contact information of the responsible researcher. The parent then chooses which child will be participating in this study; if the child is outside the age range, a warning is displayed indicating that data collected may not be used in the study, and encouraging the family to wait to participate if the child is too young.

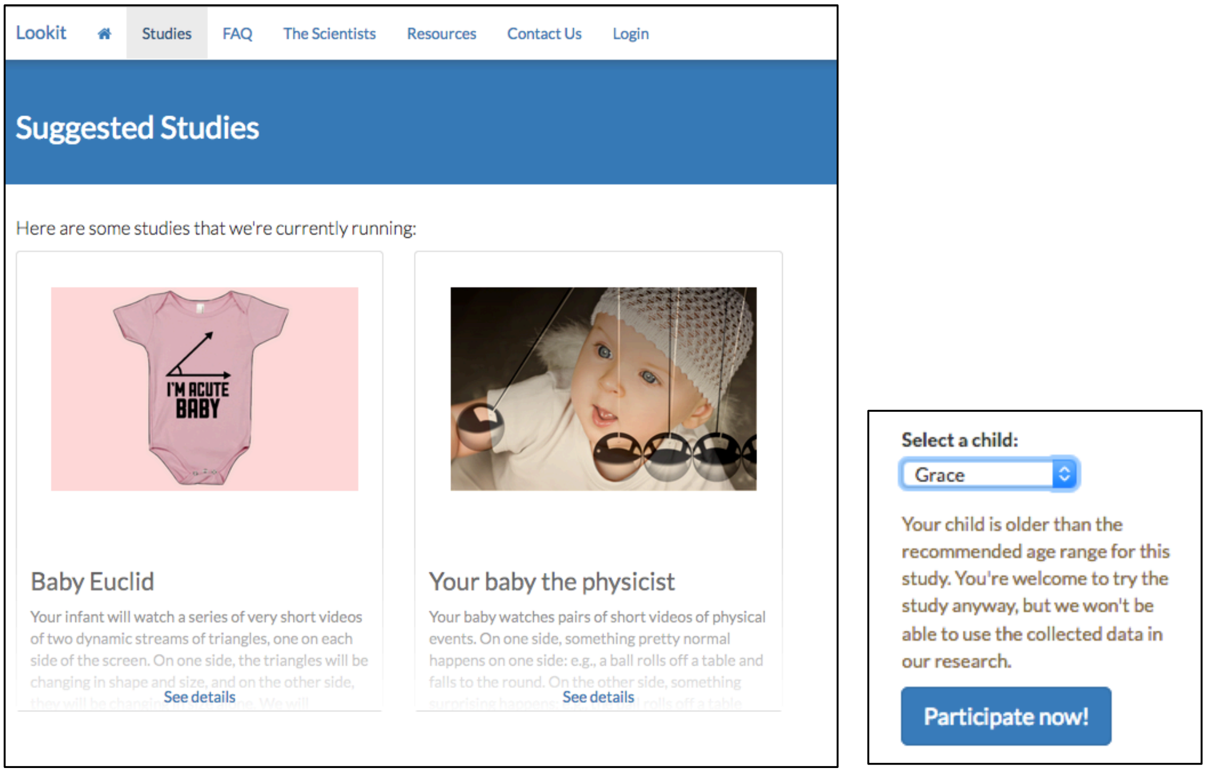



Figure 5.2. “Studies” page showing current studies on Lookit (left) and example warning displayed when the selected child is out of the age range for the selected study (right).

Upon starting a particular study, the family will first be guided through webcam setup and making a verbal statement of consent (Figure 5.3). They can stop the study at any point by pressing F1, which brings them to the exit survey at the end of the study. During the exit survey parents confirm the child’s birthdate (to check for typos during registration, or accidental selection of the wrong child), select a privacy level for their video data (viewable by Lookit researchers only, sharing allowed for scientific or educational purposes, or publicity use also allowed), decide whether their data may be shared on Databrary (https://www.databrary.org), and enter any comments. Video privacy settings are determined at the end of the session so parents already know whether, e.g., anything embarrassing happened during the session. Parents also have the option at this point to withdraw all video data from the study if needed.

A list of previous times the family has participated in studies is available under Studies -> Past Studies, along with any feedback provided by researchers for each session. For instance, researchers might thank each participant personally and let them know that data collection went well, let participants know about any webcam problems that could be fixed for a future session, or provide metrics of interest about the child's responses.

The Lookit website also includes pictures and short bios of researchers; a directory of in-person developmental labs in the USA by state, which is the first such directory to our knowledge; several activities parents can try at home to learn more about cognitive development, with video demonstrations created during an IAP class; and answers to frequently asked questions about participating.

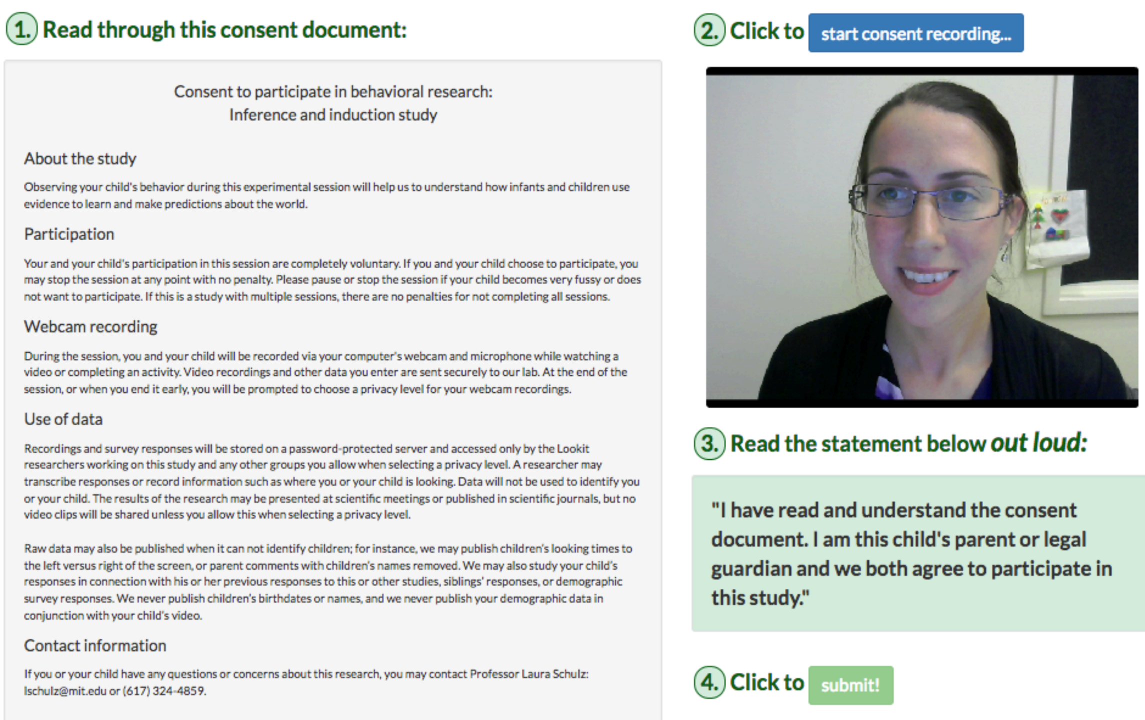



Figure 5.3. Providing verbal consent to participate on Lookit. The consent form is displayed at the left, and parents follow the step-by-step directions to record a video of themselves reading the statement out loud.

**Researcher experience.** Researchers log in to Experimenter using an Open Science Framework (OSF) ID. Upon logging in to Experimenter for the first time, they are visible to Lookit "organization admins," who can optionally give them permissions to view and/or edit all other studies (Figure 5.4). By default a researcher can only view and edit details for a study he/she created or was invited to collaborate on by the study creator or an organization admin.


Figure 5.4. Lookit organization admin view of researcher details, allowing the admin to set permissions for this researcher.

Any researcher can create new studies on Experimenter; however, these studies are not automatically shown on Lookit. After providing the participant-facing description, defining the study procedure, and testing the study, the researcher submits the study for approval, which "freezes" its state. If the study is approved by a Lookit organization admin (see Figure 5.5), the researcher then has the ability to start and stop the study (automatically and immediately making it accessible/inaccessible to participants on Lookit), but further changes require re-approval. This will support any necessary limits on researchers' use of Lookit; in the future, an admin can check that the study works, follows any site policies, and has IRB approval from the appropriate institution before granting approval. Approval will also allows manual implementation of mechanisms for researchers to share fairly in the maintenance and development costs of the platform. At the time a study is submitted for approval, all of the code used in that study is "bundled" together and will not change further, even if some of the components used in the study are updated during data collection, to protect the intended function of the study and replicability of results.

Studies may be designated as "discoverable," meaning that they will be publicly available on Lookit, or non-discoverable, in which case a link to the study is generated and can be provided directly to participants. Non-discoverable studies may be useful, for instance, when conducting an online behavioral follow-up after an in-lab fMRI session or when recruiting only participants from a particular population.

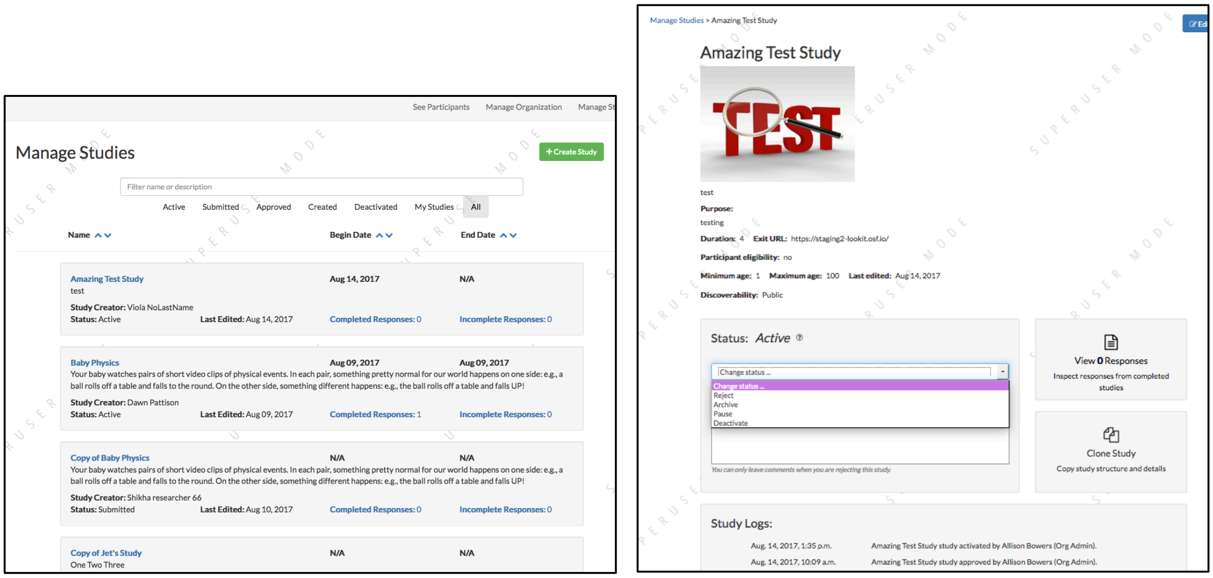



Figure 5.5. Lookit organization view of current studies, sorted by status (left) and of a single submitted study (right) with options to approve or reject.

Participant data (username, demographic form, registered children) are available only for registered participants who participated in this study; that is, a researcher can only see any data from participants in at least one of his/her studies. This simplifies future IRB considerations for handling participant data shared among labs.

Researchers do not have direct access to any participant email addresses, in order to allow programmatic enforcement of contact settings selected by participants. However, a researcher can send an email to any participant in one of their studies (which will be sent unless the participant has opted out of all contact) or send an email to all participants in the study (e.g. to announce study results) using the Experimenter interface.

**Admin interface.** A top-level admin can log in to the Django administration interface at [https://staging2-lookit.osf.io/admin/](https://staging2-lookit.osf.io/admin/) (see Figure 5.6) to view/edit all studies, researcher accounts, participant accounts, and static website content (e.g. news or FAQs).

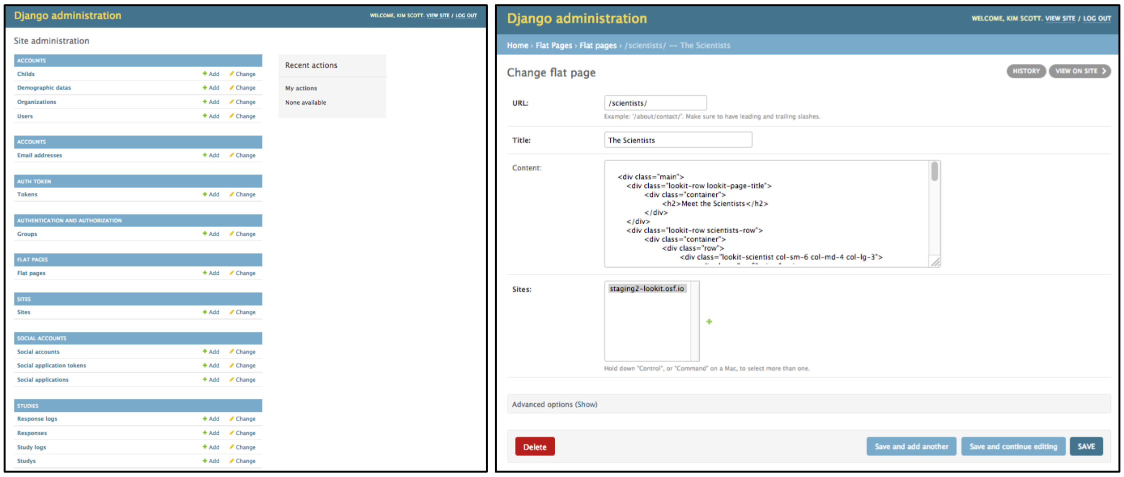



Figure 5.6. Django administration interface for Lookit/Experimenter, showing the ability to view all participant, session, study, and researcher data (overview, left) and edit static site content (right).

**Staging servers.** "Staging" instances of both Lookit and Experimenter are available, providing a sandbox environment where developers can try out changes to the platform before changing the production servers and collaborators practice writing and administering studies.

**Stack.** A variety of technologies support the functionality described. For reference, these include: Django, the Django REST framework, Django allauth (for researcher authentication via OSF), Django Guardian for object permissions, python3, celery (for queuing tasks/jobs), docker containers to deploy studies, pytransitions (https://github.com/pytransitions/transitions) for study state transitions, Ember for experiment components, the Google cloud platform for hosting, Google cloud storage for assets, Wowza for video streaming (running on an Amazon EC2 instance), HDFVR for the Flash webcam recording application, and SendGrid for email permissions.

**Documentation.** In addition to documentation for developers included in the lookit, experimenter, and exp-addons Github repos, documentation about how to create a study is available for researchers at http://lookit.readthedocs.io/en/latest/. Documentation of specific experimental components ("frames") is generated automatically using YUIdoc from comments in the code to maximize maintainability. This documentation is available at http://centerforopenscience.github.io/exp-addons/modules/frames.html.

## 5.2 Designing and running a study

**Specification of study protocol using JSON schema.** Researchers specify the protocol for a Lookit study by providing a JSON (**J**ava**S**cript **O**bject **N**otation) object on the Experimenter interface, which is interpreted according to a JSON Schema (http://json-schema.org/) designed for Lookit studies. A JSON schema describes a class of JSON objects, indicating what type of data to expect and require.

No programming is required to design a study: JSON is a simple, human-readable text format for describing data (see http://www.json.org/). A JSON object is an unordered set of key – value pairs, written as follows:

- The object itself is enclosed in curly braces.
- Keys are unique strings enclosed in double quotes.
- A key and value are separated by a colon.
- Key-value pairs are separated by commas.

A JSON *value* can be any of the following: a string (enclosed in double quotes), a number, a JSON object (as described above), an array (an ordered list of JSON values, separated by commas and enclosed by square brackets), true, false, or null. There are no requirements for specific formatting of a JSON document (any whitespace not part of a string is ignored). Here is an example JSON object to illustrate these principles:

```
{
	“name”: “Jane”,
	“age”: 43,
	“favoritefoods”: [
		“eggplant”,
		“apple”,
		“lima beans”
	],
	“allergies”: {
		“peanut”: “mild”,
		“shellfish”: “severe”
	}
}
```

The *keys* are the strings “name”, “age”, “favoritefoods”, and “allergies”. Favorite foods are stored as an array, or ordered list; allergies are stored as a JSON object mapping food names to severity of reaction. The same object could also be written as follows:

```
{“age”: 43, “allergies”: {“peanut”: “mild”, “shellfish”: “severe”}, “name”:
“Jane”, “favoritefoods”: [“eggplant”, “apple”, lima beans”]}
```

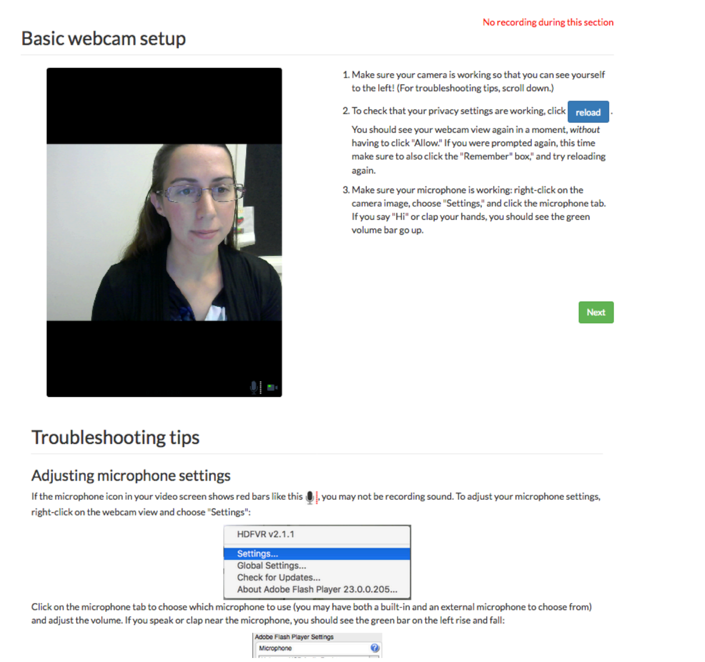
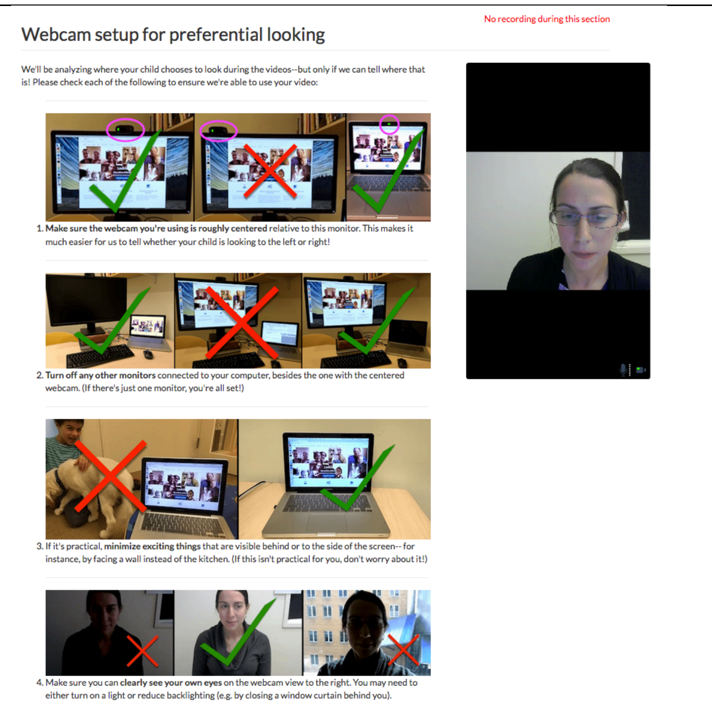
Studies on Lookit are broken into a set of fundamental units called “frames,” which can also be thought of as “pages” of the study. A single experimental trial (e.g. looking time measurement) would generally be one frame, as are the video consent procedure (shown in Figure 5.3) and exit survey. The JSON Schema used to parse Lookit studies requires a JSON object with two keys: “frames” and “sequence”. The “frames” value defines the frames used in this study: it must be a JSON object mapping frame nicknames (any unique strings chosen by the researcher) to frame objects (defined next). The “sequence” value must be an ordered list of the frames to use in this study; values in this list must be frame nicknames from the “frames” value. Here is the JSON for a very minimal Lookit study:

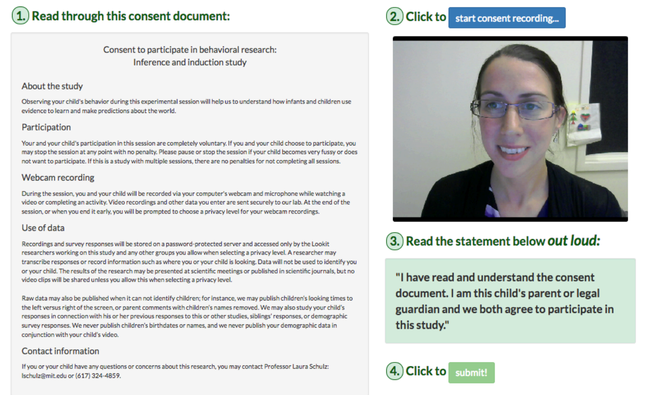
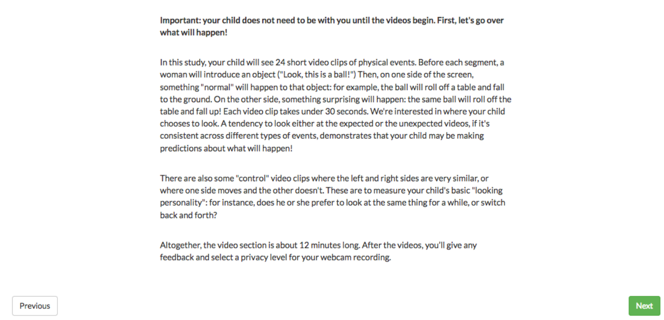
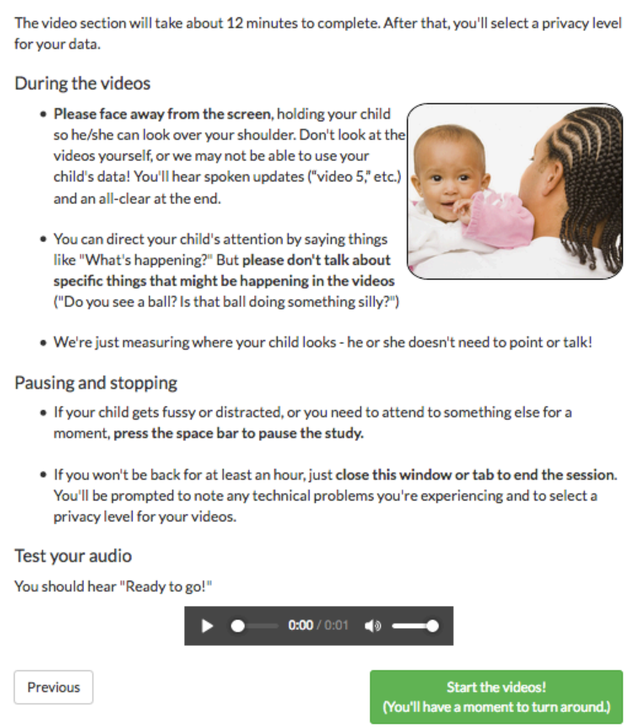
```
{
      “frames”: {
            “my-consent-frame”: {
                  “kind”: “exp-video-consent”,
                  “prompt”: “I agree to participate”,
                  “blocks”: [
                        {
                              “title”: “About the study”,
                              “text”: “This isn’t a real study.”
                        }
                  ]
            },
            “my-exit-survey”: {
                   “kind”: “exp-lookit-exit-survey”,
                   “debriefing”: {
                        “title”: “Thank you!”,
                        “text”: “You participated.”
                   }
            }
        },
      “sequence”: [
            “my-consent-frame”,
            “my-exit-survey”
      ]
}
```

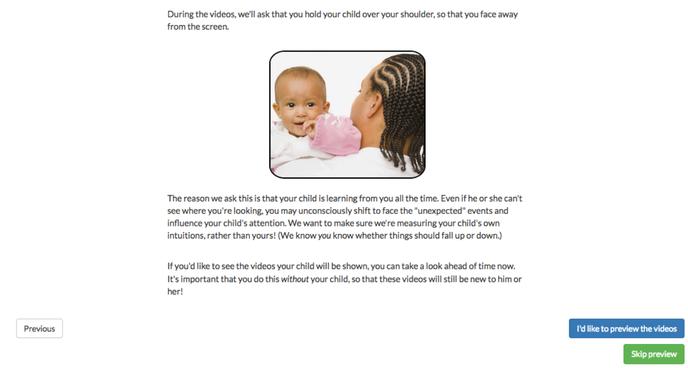
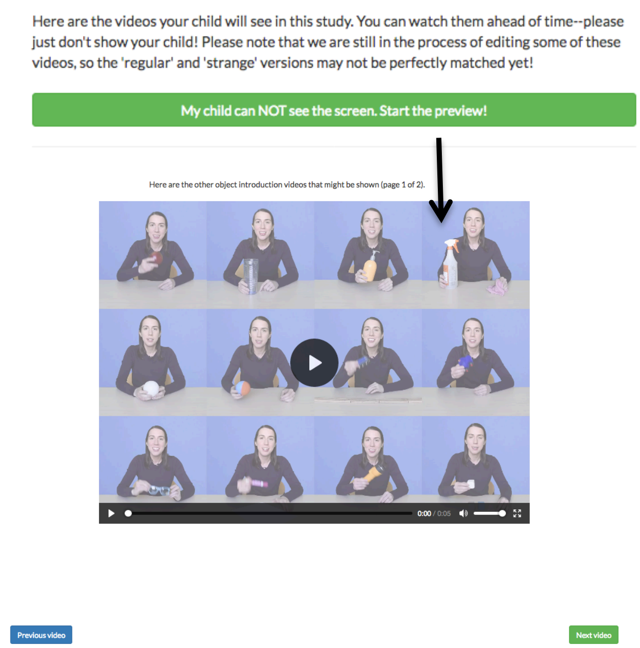
This JSON specifies a Lookit study with two frames, consent and an exit survey. Note that the frame nicknames “my-consent-frame” and “my-exit-survey” that are defined in “frames” are also used in the “sequence”. Frames may be specified but not used in “sequence”. Within each frame object (highlighted in gray above), a “kind” must be specified. This determines the frame type that will be used. Additional data may be included in the frame object to customize the behavior of the frame, for instance to specify instruction text or the stimuli to use for a test trial. The keys that may (or must) be included in a frame object are determined by the frame type; each frame definition includes a JSON Schema describing the expected data to be passed. Multiple frames of the same kind may be included in a study – for instance, test trials using different stimuli.

The separation of frame definitions and sequence allows researchers to easily and flexibly edit and test study protocols – for instance, the order of frames may be altered or a particular frame removed for testing purposes without altering any frame definitions.

**General-use frames available.** Full documentation of the frames available for use on Lookit is available at http://centerforopenscience.github.io/exp-addons/modules/frames.html. For each frame, a general description with example is provided, along with a listing (under “Properties”) of all key/value pairs that can be used to customize frame behavior and a listing of the types of events recorded during this frame to be included in session data (under “Events”). Commonly used frames are described in Table 5.1.

<table>
<tr><th>Kind</th><th>Description</th><th>Example screenshot</th></tr>
<tr><td>exp-video-config</td><td>Displays the user’s webcam and a sequence of steps to ensure the webcam is working and permissions are set properly; includes troubleshooting tips. Not customizable.</td><td>
</td></tr>
<tr><td>exp-video-config-quality</td><td>Displays the user’s webcam along with a sequence of issues to check for to improve quality of the video collected (center webcam, turn off additional monitors, minimize distractions, make sure eyes are visible). Not customizable.</td><td>
</td></tr>
</table>

<table>
<tr>
<td>exp-video-consent</td>
<td>Displays a consent document and the user’s webcam; user clicks a button to begin consent recording, reads a statement aloud, and clicks “Done” to proceed. Title and text of consent document and text of statement to read are customizable. Records video between clicking “Record” and “Done.”</td>
<td>
</td>
</tr>
<tr>
<td>exp-lookit-text</td>
<td>Displays text to the participant, e.g. to show instructions or a summary of the experiment. Researcher specifies a list of paragraphs with optional titles and whether to show a “previous” button to return to the last frame.</td>
<td>
</td>
</tr>
<tr>
<td>exp-lookit-instructions</td>
<td>Displays bulleted text to participant and includes audio controls for an audio clip. Researcher specifies array of bulleted text lists with titles (bullet points may include images), text of “next” button, optional audio source, and whether audio must be played to proceed. This frame is intended to show instructions immediately before proceeding to test trials, including a check that the participant’s speakers are on.</td>
<td>
</td>
</tr>
</table>

<table>
<tr>
<td>exp-lookit-preview-explanation</td>
<td>Displays text and an optional image to participant, along with buttons that allow the participant to either proceed to the next frame or skip one frame. Text blocks, image source, button text, and whether to show a ‘previous’ button are customizable. This is intended for offering parents the opportunity to preview stimuli used in the study if desired (a preview would then be the next frame).</td>
<td>
</td>
</tr>
<tr>
<td>exp-video-preview</td>
<td>First displays text with a button to proceed, then displays a sequence of videos (with controls) and/or images with captions, which the participant can proceed through using next/previous buttons. Video recording starts after clicking the proceed button, and ends at the end of the frame. Initial text, button text, videos, and captions are customizable. This is intended to allow to preview the stimuli used in the study, and checking that the child is not also watching.</td>
<td>
</td>
</tr>
</table>

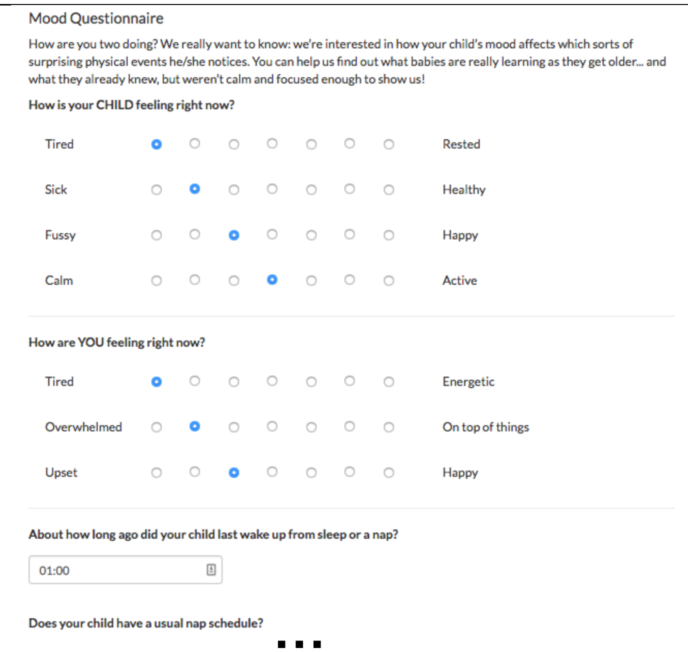
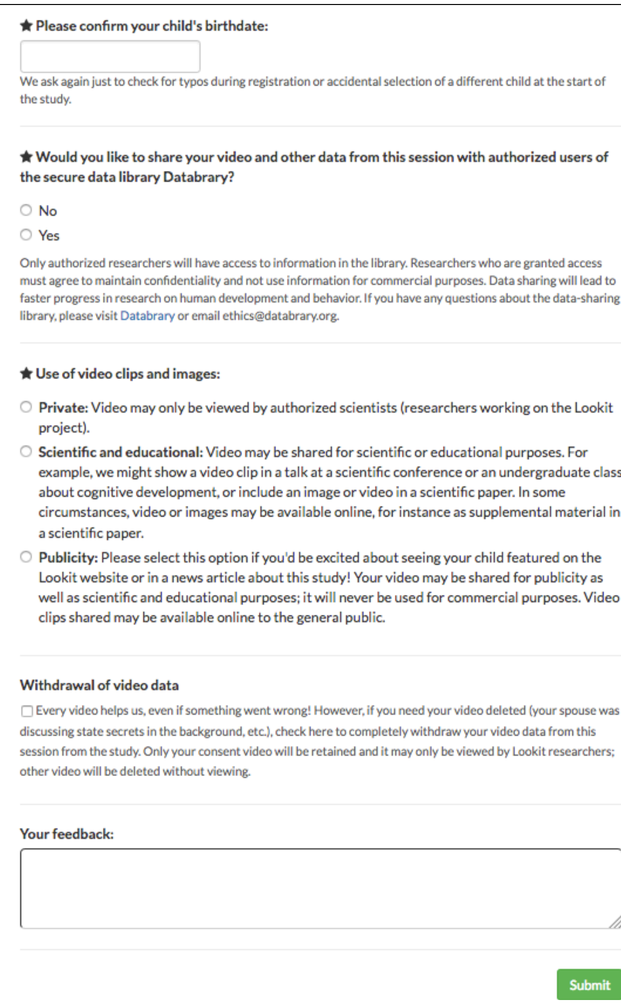
| | | |
|---|---|---|
| exp-lookit-mood-questionnaire | Displays a simple mood questionnaire for an infant/child and parent, with input validation (ensuring that all questions are answered). Only intro text is currently customizable; this is intended as a base for future development of a fully customizable survey frame. |  |
| exp-lookit-exit-survey | Displays a standard exit survey to use at the end of Lookit studies: asks participants to confirm child's birthdate, whether to share data with Databrary, privacy level of videos, and give any general feedback. Input validation ensures all questions are answered. After submitting the survey, customizable debriefing text is shown. |  |

Table 5.1: Available Lookit experiment frames for general use.

**Providing and using custom code.** Instead of using the frames available on Lookit (via the COS exp-addons repo), researchers may also specify a different Github repository with their custom-written frame definitions, and continue to use the online interface for JSON specification and study testing. We expect the most common use case to be researchers creating a fork of COS/exp-addons and adding several of their own custom frames to support a specific experiment. Information about developing new custom frames is available as part of the Experimenter documentation (experimenter.readthedocs.io); frames are defined using the Ember.js front-end framework (https://www.emberjs.com/).

This option allows researchers with sufficient time and expertise to do custom development without central support from the Lookit project. Allowing use of a separate repository avoids the need for a central authority to promptly review requested additions to the central codebase (and manage any conflicts) in order for labs to use their new frames, as well as the danger of adding to a central repository frames that are hyper-specific or poorly documented. It also supports researchers who wish to share code they have developed with others: for instance, one lab could independently maintain a fork of exp-addons specialized for reaction-time studies and other researchers could use (and cite) that fork.

## 5.3 Condition assignment and counterbalancing

Most experimental designs, of course, involve more than a single exact protocol: participants may be assigned to different experimental conditions, and even within conditions, potentially confounding variables (such as order of test trials or placement of stimuli) may be counterbalanced. On Lookit, special frame types called randomizers support such designs. A randomizer frame is expanded into a sequence of frames; the particular frames and their order may be randomized, according to the function of the particular randomizer frame.

To use a randomizer rather than another type of frame, set “kind” to “choice” and “sampler” to the appropriate randomizer name. We will focus here on the most commonly-used and general randomizer type, called “random-parameter-set”; for documentation of all available randomizers, see http://centerforopenscience.github.io/exp-addons/modules/randomizers.html. To select this randomizer, your frame object must have the appropriate "kind" and "sampler":

```
{
        "sampler": "random-parameter-set",
        "kind": "choice",
        ...
```

```
}
```

In addition, there are three special properties you need to define to use random-parameter-set: “frameList”, “commonFrameProperties”, and “parameterSets”. frameList is just what it sounds like: a list of all the frames that should be generated by this randomizer. Each frame is a regular frame object as would be used for a standalone frame, with two differences:

- Default properties, to share across all of the frames generated by this randomizer unless overwritten in a particular frame, may be defined in the JSON object commonFrameProperties instead, as a convenience.
- Placeholder strings may be used in place of any of the properties in the frame; they will be replaced based on the values in the selected parameterSet.

parameterSets is a list of mappings from placeholder strings to actual values. When a participant starts the study, one of these sets will be randomly selected, and any parameter values in the frameList (including commonFrameProperties) that match any of the keys in this parameter set will be replaced.

We first walk through a simple example of using this randomizer. Suppose we start with the following JSON document describing a study that includes instructions, an experimental manipulation asking participants to think about how delicious broccoli is, and an exit survey.

```
{
	"frames": {
		"instructions": {
			"id": "text-1",
			"blocks": [{
				"text": "Some introductory text about this study."
			}, {
				"text": "Here's what's going to happen! You’re going
to think about how tasty broccoli is.”
			}],
			"showPreviousButton": false,
			"kind": "exp-lookit-text"
		},
		"manipulation": {
			"id": "text-2",
			"blocks": [{
				"text": "Think about how delicious broccoli is."
			}, {
				"text": "It is so tasty!"
			}],
			"showPreviousButton": true,
			"kind": "exp-lookit-text"
		},
		"exit-survey": {
			"debriefing": {
				"text": "Thank you for participating in this study!
",
```

```
                        "title": "Thank you!"
                    },
                    "id": "exit-survey",
                    "kind": "exp-lookit-exit-survey"
            }
    },
    "sequence": ["instructions", "manipulation", "exit-survey"]
}
```

What we actually want, however, is for some participants to think about how tasty broccoli is, and others to think about how yucky it is! We can use a random-parameter-set frame to replace both text frames:

```
{
    "frames": {
        "instruct-and-manip": {
            "sampler": "random-parameter-set",
            "kind": "choice",
            "id": "instruct-and-manip",
            "frameList": [
                {
                    "blocks": [
                        {
                            "text": "Some introductory text
    about this study."
                        },
                        {
                            "text": "INTROTEXT"
                        }
                    ],
                    "showPreviousButton": false
                },
                {
                    "blocks": [
                        {
                            "text": "MANIP-TEXT-1"
                        },
                        {
                        "text": "MANIP-TEXT-2"
                        }
                    ],
                    "showPreviousButton": true
                }
            ],
            "commonFrameProperties": {
                "kind": "exp-lookit-text"
            },
            "parameterSets": [
                {
                    "INTROTEXT": "Here's what's going to happen!
    You're going to think about how tasty broccoli is.",
                    "MANIP-TEXT-1": "Think about how delicious
    broccoli is.",
                    "MANIP-TEXT-2": "It is so tasty!"
                },
                {
```

```
                              "INTROTEXT": "Here's what's going to happen!
       You're going to think about how disgusting broccoli is.",
                              "MANIP-TEXT-1": "Think about how disgusting
       broccoli is.",
                              "MANIP-TEXT-2": "It is so yucky!"
                        }
                  ]
            },
            "exit-survey": {
                  "debriefing": {
                        "text": "Thank you for participating in this study!
",
                        "title": "Thank you!"
                  },
                  "id": "exit-survey",
                  "kind": "exp-lookit-exit-survey"
            }
      },
      "sequence": ["instruct-and-manip", "exit-survey"]
 }
```

Notice that since both of the frames in the frameList were of the same kind, we could define the kind in commonFrameProperties. We no longer define id values for the individual frames, as they will be automatically identified as instruct-and-manip-1 and instruct-and-manip-2.

To evaluate the “instruct-and-manip” randomizer is evaluated, the Lookit experiment player will start with the frameList and add the key-value pairs in commonFrameProperties to each frame (not overwriting existing pairs):

```
[
      {
            “kind”: “exp-lookit-text”,
            "blocks": [
                  {
                        "text": "Some introductory text about this study."
                  },
                  {
                        "text": "INTROTEXT"
                  }
            ],
            "showPreviousButton": false
      },
      {
            “kind”: “exp-lookit-text”,
            "blocks": [
                  {
                        "text": "MANIP-TEXT-1"
                  },
                  {
                        "text": "MANIP-TEXT-2"
                  }
            ],
            "showPreviousButton": true
      }
]
```

Next, one of the two objects in parameterSets is selected randomly. By default, parameter sets are weighted equally, but parameterSetWeights can be provided as an optional key in the random-parameter-set frame. If provided, parameterSetWeights should be an array of relative weights for the parameter sets, corresponding to the order they are listed. For instance, if we wanted 75% of participants to think about how tasty broccoli is, we could set parameterSetWeights to [3, 1]. This allows uneven condition assignment where needed to optimize power, as well as allowing researchers to stop testing conditions that already have enough participants as data collection proceeds. Suppose that in this case the second parameter set is selected:

```
{
        "INTROTEXT": "Here's what's going to happen! You're going to think
       about how disgusting broccoli is.",
        "MANIP-TEXT-1": "Think about how disgusting broccoli is.",
        "MANIP-TEXT-2": "It is so yucky!"
 }
```

Now we return to the list of frames, and wherever any value matches one of the keys in the parameterSet (even if that value is nested in another object), it is replaced by the corresponding value from the parameterSet, yielding the following final list of frames:

```
[
      {
            “kind”: “exp-lookit-text”,
            "blocks": [
                  {
                        "text": "Some introductory text about this study."
                  },
                  {
                        "text": "Here's what's going to happen! You're going
            to think about how disgusting broccoli is."
                  }
            ],
            "showPreviousButton": false
      },
      {
            “kind”: “exp-lookit-text”,
            "blocks": [
                  {
                        "text": "Think about how disgusting broccoli is."
                  },
                  {
                        "text": "It is so yucky!"
                  }
            ],
            "showPreviousButton": true
      }
```

```
]
```

In more complex experimental designs, the frames created by a randomizer may themselves be randomizers! This nesting allows more modular specification: for instance, a study might have ten test trials, each of which consists of three phases. The "outer" randomizer could then generate a frameList of ten randomizer frames, each of which would be resolved in turn into three frames. Below is a simplified example with only two test trials, each of which has three phases:

```
{
      "sampler": "random-parameter-set",
      "frameList": [
        {
          "parameterSets": [
             {
               "NTRIAL": 1,
               "PHASE1STIM": "T1P1",
               "PHASE2STIM": "T1P2",
               "PHASE3STIM": "T1P3"
             }
          ]
        },
        {
          "parameterSets": [
             {
               "NTRIAL": 2,
               "PHASE1STIM": "T2P1",
               "PHASE2STIM": "T2P2",
               "PHASE3STIM": "T2P3"
             }
          ]
        }
      ],
      "parameterSets": [
            {
                  "T1P1": "mouse",
                  "T1P2": "rat",
                  "T1P3": "chipmunk",
                  "T2P1": "horse",
                  "T2P2": "goat",
                  "T2P3": "cow"
            },
            {
                  "T1P1": "guppy",
                  "T1P2": "tadpole",
                  "T1P3": "goldfish",
                  "T2P1": "whale",
                  "T2P2": "manatee",
                  "T2P3": "shark"
            }

      ],
      "commonFrameProperties": {
```

```
            "sampler": "random-parameter-set"
            "frameList": [
                    {
                          "nPhase": 1,
                          “animal”: “PHASE1STIM”
                    },
                    {
                          "nPhase": 2,
                          “animal”: “PHASE2STIM”
                    },
                    {
                          "nPhase": 3,
                          “animal”: “PHASE3STIM”
                    }
            ],
            "commonFrameProperties": {
              "nTrial": "NTRIAL",
              "kind": "question-about-animals-frame"
            }
        }
}
```

To evaluate this experiment frame, the Lookit experiment player starts with the list of frames in **frameList**, adding the key:value pairs in **commonFrameProperties** to each frame, which yields the following list of frames (color of keys matches that of their origin above):

```
[
        {
            "parameterSets": [
                    {
                          "NTRIAL": 1,
                          “PHASE1STIM”: “T1P1”,
                          “PHASE2STIM”: “T1P2”,
                          “PHASE3STIM”: “T1P3”
                     }
                ],
            "sampler": "random-parameter-set"
            "frameList": [
                    {
                          "nPhase": 1,
                          “animal”: “PHASE1STIM”
                    },
                    {
                          "nPhase": 2,
                          “animal”: “PHASE2STIM”
                    },
                    {
                          "nPhase": 3,
                          “animal”: “PHASE3STIM”
                    }
            ],
            "commonFrameProperties": {
                    "nTrial": "NTRIAL",
                    "kind": "question-about-animals-frame"
            }
```

```
        },
        {
            "parameterSets": [
                {
                        "NTRIAL": 2,
                        "PHASE1STIM": "T2P1",
                        "PHASE2STIM": "T2P2",
                        "PHASE3STIM": "T2P3"
                }
            ],
            "sampler": "random-parameter-set"
            "frameList": [
                {
                        "nPhase": 1,
                        "animal": "PHASE1STIM"
                },
                {
                        "nPhase": 2,
                        "animal": "PHASE2STIM"
                },
                {
                        "nPhase": 3,
                        "animal": "PHASE3STIM"
                }
            ],
            "commonFrameProperties": {
                "nTrial": "NTRIAL",
                "kind": "question-about-animals-frame"
            }
        }
]
```

One of the two **parameterSets** is then selected randomly; suppose the second one (aquatic instead of land animals) is selected. Now any substitutions are made based on the keys in this parameterSet. The first frame in the sequence is now:

```
        {
            "parameterSets": [
                {
                        "NTRIAL": 1,
                        "PHASE1STIM": "guppy",
                        "PHASE2STIM": "tadpole",
                        "PHASE3STIM": "goldfish"
                  }
                ],
            "sampler": "random-parameter-set"
            "frameList": [
                {
                        "nPhase": 1,
                        "animal": "PHASE1STIM"
                },
                {
                        "nPhase": 2,
                        "animal": "PHASE2STIM"
```

```
            },
            {
                  "nPhase": 3,
                  "animal": "PHASE3STIM"
            }
      ],
      "commonFrameProperties": {
            "nTrial": "NTRIAL",
            "kind": "question-about-animals-frame"
      }
  }
```

Next, each frame is expanded since it is in turn another randomizer (due to "sampler": "random-parameter-set"). The frame above, representing Trial 1, will be turned into three frames. First, again, we start with the frameList, and merge the commonFrameProperties into each frame:

```
 [
      {
            "nPhase": 1,
            "animal": "PHASE1STIM",
            "nTrial": "NTRIAL",
            "kind": "question-about-animals-frame"
      },
      {
            "nPhase": 2,
            "animal": "PHASE2STIM",
            "nTrial": "NTRIAL",
            "kind": "question-about-animals-frame"
      },
      {
            "nPhase": 3,
            "animal": "PHASE3STIM",
            "nTrial": "NTRIAL",
            "kind": "question-about-animals-frame"
      }
]
```

Finally, a parameter set is selected from parameterSets. Only one parameter set is defined for this trial, which is deliberate; it simply selects the correct stimuli for this trial. Substituting in the values from the parameter set yields the following frames:

```
[
      {
            "nPhase": 1,
            "animal": "guppy",
            "nTrial": 1,
            "kind": "question-about-animals-frame"
      },
      {
            "nPhase": 2,
            "animal": "tadpole",
            "nTrial": 1,
```

```
            "kind": "question-about-animals-frame"
      },
      {
            "nPhase": 3,
            “animal”: “goldfish”,
            "nTrial": 1,
            "kind": "question-about-animals-frame"
      }
]
```

The random-parameter-set randomizer is expected to be general enough to capture most experimental designs that researchers put on Lookit, but additional custom randomizers can also be designed for specific studies or to provide simpler syntax for common use cases.

## 5.4 Video recording

Webcam video recording during Lookit frames is accomplished using Flash as the interface to the webcam, and HDFVR (https://hdfvr.com/) for Javascript and server-side APIs that support options such as setting preferred audio and video quality and codecs. Recorded video is streamed using a Wowza Streaming Engine running on an Amazon EC2 instance and stored temporarily on EC2 before being copied to Amazon S3 storage.

Lookit frames that collect video data make use of an Ember mixin ‘VideoRecord’ included in exp-addons, which makes a VideoRecorderObject available for use in the code for that frame. This object includes methods for showing/hiding the webcam view, starting/pausing/resuming/stopping video recording, installing/destroying the recorder, and checking the current video timestamp (see http://centerforopenscience.github.io/exp-addons/classes/VideoRecorderObject.html). The programmer designing a new frame can therefore flexibly indicate when recording should begin and end, as well as recording video timestamps for any events recorded during this frame (e.g., so that during later data analysis, researchers know the exact time in the video where a new stimulus was presented). The name(s) of any videos collected during a particular frame as included in the session data recorded, to facilitate matching sessions to videos; video filenames also include the study ID, session ID, frame ID, and a timestamp.

One technical challenge imposed by webcam video streaming is that a connection to the server must be established before webcam recording can be quickly turned on and off, and this process may take up to several seconds. Each experiment frame records a separate video clip and establishes a separate connection to the server, so frames must be designed to wait for recording to begin before proceeding to a portion of the trial where video data is required. This fits well with typical study designs using looking time or preferential looking, where the child's attention is returned to the center of the screen between trials; the first few seconds of the child watching the "attention grabber" are not critical and we can simply ensure that the webcam connection is established before proceeding to the actual experimental trial. When collecting verbal responses, the study frame can simply pause until the connection is established or, similarly, proceed with an initial portion of the trial where video data is not required. Currently, however, continuous webcam recording across frames is not possible on Lookit; any periods of continuous recording must be within a single frame.

## 5.5 Receiving and interpreting study data

A researcher with edit permissions for a particular study can download session data in JSON or CSV format via the Experimenter interface. A session record in a Postgres database is created each time a participant starts the study, and includes a timestamp, account information, condition assignment, the sequence of frames the participant actually saw, and frame-specific information for each frame (included in an 'expData' structure which is a JSON object with keys corresponding to frame nicknames, described in more detail at experimenter.readthedocs.io/en/latest/experiments.html). Each frame type may save different data, e.g. form responses; frames that record webcam video include the video filename(s). The data captured by a particular frame are listed in the frame documentation at http://centerforopenscience.github.io/exp-addons, under 'Methods' > 'serializeContent'. Additionally, event data is captured for each frame and included under an eventTimings key within the frame data JSON, minimally including a timestamped event when the user proceeds to the next frame. These events are listed under 'Events' in the documentation.

Webcam videos of participants in the study may be downloaded individually or a .zip archive containing all study video may be requested (because the latter file may be very large, the task is handled asynchronously and the researcher receives an email when the file is available). Alternately, the researcher can download all consent videos only, which is intended to support labs in developing reliable workflows for checking for informed consent before proceeding to other video coding.

## 5.6 Studies implemented

### 5.6.1 Overview of development process for test studies

As development of the new platform proceeded, we began a tiered process to implement “test” studies on Lookit. These studies served as example use cases to drive feature development and refine our understanding of researchers’ requirements, while beginning to conduct genuine novel research on the platform.

The first study we implemented was our own longitudinal study of infants’ intuitive physics judgments (Study 1). This relatively complex protocol showcases Lookit’s unique potential to support densely repeated measurements, and its stringent requirements motivated much of the initial functionality needed for an early single-lab version of the new platform (before appropriate permissions for multiple researchers were implemented).

Next, we began “round 1” of beta testing by inviting a study from a collaborating institution (Study 2). We fully implemented this study based on specifications provided by and discussions with the other lab, writing custom experimental frames as needed and providing the JSON document describing the study. Data collection proceeded using the single-lab system, with the appropriate videos and data processed by MIT and provided to the collaborating institution. Development for this round can be found in the ‘geometry’ branch of exp-addons: https://github.com/kimberscott/exp-addons/tree/geometry.

Finally, in “round 2” of beta testing we solicited several studies for which we implemented all necessary custom experimental frames, but the researchers involved agreed to create the stimuli needed and write the JSON documents for their studies starting from a provided scaffold demonstrating use of custom frames & randomizers. Studies were written and tested on a staging server (“sandbox”) to which all beta testers received access, with MIT responsible for moving studies to the production site when ready. We consulted on the adaptation of in-lab studies to the online environment (e.g., how and when to include calibration procedures, how to minimize parent interference). This allowed us to try out study development procedures and inform the design of the final Experimenter user interface. We selected several studies from potential collaborators who had a reasonably simple developmental study already prepared and a researcher with a strong programming background as well as some familiarity with Javascript/CSS/HTML. As per the current state of the platform, we recruited replications of in-lab studies where researchers hoped to bring the general protocol online eventually (perhaps allowing parametric variation), rather than ambitious new designs or high-pressure requirements for rapid recruitment. Development for this round can be found in the 'round2' branch of exp-addons: https://github.com/kimberscott/exp-addons/tree/round2.

All beta testing was conducted under an MIT IRB protocol, with collaborating researchers responsible for recruiting participants for their own studies; we also facilitated collaborative recruitment efforts among labs. Details of potential studies to put on the platform were solicited using an evolving standard request describing the information needed: a rough list of "frames," the layout and behavior needed for test trials, and example or placeholder stimuli to use for an initial mockup. A detailed request with examples was necessary because the level of detail required to programmatically implement a study protocol goes beyond than that provided in a typical methods section, and the sorts of information that must be specified (e.g., what should happen if the user leaves fullscreen mode; exactly how condition assignment determines the sequence of trials a participant sees; how a layout should scale on monitors of different aspect ratios) is not intuitively clear to researchers used to designing in-person studies.

The main challenge during custom frame development was maximizing the generalizability of new frames without sacrificing usability for researchers. At some point, of course, it becomes easier to simply program a bespoke version of a frame—using a known programming language intended to allow specification of arbitrary functionality—rather than to interact with a giant set of potential parameters in JSON. For each of these test studies, we created frames that were sufficiently generalizable to allow use across all trials of a particular study, with researchers able to change any parameters they might want to adjust on their own (e.g. wording, formatting, stimulus choice). However, we expect that new studies will require creation of slightly altered custom frames and/or expanded functionality. Generalizing gradually based on demand, while maintaining good general practices in e.g. building abstraction layers, allows a better fit between the available frame options and what researchers actually want to do.

### 5.6.2 Study 1: Dense sampling of intuitive physics judgments from individual infants

**Overview.** In this study—our own first on the new platform—we aim to assess variation both between and within infants in their expression of understanding of several physical principles (gravity, inertia, and support) via preferential looking. We are interested in the stability of infants' preferences for violations of these principles across illustrative event types (e.g., objects rolling up ramps vs. falling up off of tables), other stimulus variation (e.g., varied objects, camera angles, backgrounds, and left/right placement), and mood and attentional states, as well as the stability and predictive value of control measures of children's more general preferential looking behavior.

Children complete 24 preferential looking trials per study session (about 12 minutes), and families are encouraged to complete 15 sessions within 2 months – an unprecedented amount of preferential looking data from individual, typically-developing infants. Parents complete a short mood survey (including questions about the child's attentional and mood state, as well as when he/she last ate and slept) and review instructions before the preferential looking portion.

Each preferential looking trial begins with an object that lasts about 5 seconds, intended as an attention-getter to re-orient children towards the center while in principle reinforcing that the object in question is not an agent and should be expected to follow normal physical laws. Then two events involving that object are shown simultaneously, one on the left and one on the right, looping continuously for 24s (event videos range from about 2-5s). Events always show the same object, same camera angle, same background, with a difference only in the "outcome." Event types are shown Table 5.2; there are a total of 1264 unique side-by-side videos used in the study. Parents can pause individual trials. If they pause during the intro, they just start over upon restarting. If they pause during the test (up to one time per trial) they restart from the intro, but then the left and right test videos are flipped for the test phase.

| **Concept** | **Event** | **Outcome** | **Objects** | **Camera angles/BGs** | **Times shown per session** |
|---|---|---|---|---|---|
| **Gravity** | **Table**: Object is rolled/slid off a table and continues… | **down** | Apple, cup, lotion, orangeball, whiteball | 2/1 | 2 |
| | | **up** | | | |
| | | **horizontal** | | | |
| | **Toss**: Object in hand is… | Tossed up, falls **down** | Apple, cup, lotion, orangeball, spraybottle, whiteball | 2/1 | 2 |
| | | Tossed down, falls **up** | | | |
| | **Ramp**: Object is placed in center of ramp and is released to roll/slide… | **down** | Apple, cup, lotion, spraybottle, | 2/2 | 2 |
| | | **up** | | | |

| | | | | | |
|---|---|---|---|---|---|
| | | | whiteball | | |
| **Inertia** | **Stop**: Object rolls from one side of screen and stops in the middle before continuing | stopped/restarted by a **hand** | Block, flashlight, marker, sunglasses, toycar, train | 2/1 | 2 |
| | | stops/restarts on its own, with **no hand** | | | |
| | **Reverse**: Object rolls/slides from one side of the screen and collides with… | **barrier** | | 2/1 | 2 |
| | | **no barrier** (object follows same trajectory as above, but doesn't actually touch the barrier) | | | |
| **Support** | **Fall**: An object is placed […] a cabinet and immediately falls | **mostly on** | Book, brush, duck, hammer, shoe, tissues | 1/1 | 3 |
| | | **slightly on** | | | |
| | | **next to** | | | |
| | | **near** | | | |
| | **Stay**: An object is placed […] a cabinet and stays there | **mostly on** | | 1/1 | 3 |
| | | **slightly on** | | | |
| | | **next to** | | | |
| | | **near** | | | |
| **Control** | **Same**: Distinguishable but similar physically-possible human actions on objects, like rotating an object about one axis vs. another | version **A** | Box, eraser, funnel, scissors, spoon, wrench | 1/1 | 3 |
| | | version **B** | | | |
| | **Salience**: Physically-possible human actions on objects, some more interesting, like flipping a spoon vs. slowly extending it or erasing a drawing vs. an empty board | **boring** | | 1/1 | 3 |
| | | **interesting** | | | |
| | **Calibration**: spinning ball moves back and forth | [Entire screen, start left/right] | N/a | n/a | 2 |

Table 5.2: Videos shown in Study 1 to illustrate physical principles. Two "outcomes" of the same event (using the same object, camera angle, and background) are shown to children simultaneously at the right and left of the screen.

Pilot data collection for this study (43 usable sessions, one per participant) is complete, and data collection has begun. We have received 145 usable sessions to date; 5 participants out of a target of 50 have completed our goal of at least 12 usable sessions, and another 5 have completed at least 6 usable sessions.

**Custom frame.** The ‘exp-video-physics’ frame implements the functionality of a single preferential looking trial as described above. It takes parameters indicating the duration of the test trial and the URLs of the following stimuli: test video, alternate test video (used if the parent pauses during the initial preferential looking measurement), attention-grabber video and announcement audio (played at the start of each trial, e.g. to say “Trial 4 of 24”), object intro video, and music to play during the trial.

The frame implements a simple state change logic; it is always in one of three states (represented by an internal ‘currentTask’ variable): announce, intro, or test. The ‘announce’ state corresponds to the initial display of the attention-grabber video along with audio indicating the trial number, ‘intro’ to the object intro video, and ‘test’ to the preferential looking trial with looping test video. The frame can also be paused or unpaused, and the test video to use can be regular or alternate. When the frame is first loaded, it is in the ‘announce’ state, the frame is unpaused, and the test video to use is the regular test video. Transitions among states are then managed by event listeners – e.g., when the announcement audio finishes playing, the state changes to intro; when the intro video finishes playing, the state changes to test; when the user presses the space bar, the study is paused/unpaused. This general framework proved useful for studies 2 and 3 as well.

**Counterbalancing.** The counterbalancing scheme for this study is relatively complex, owing to the variety of video types to be shown and the longitudinal design. The sequence of trials to be shown is generated by a custom randomizer ‘pref-phys’. Trials cycle through gravity, inertia (plus calibration), support, and control pairings during a session; the order of these concepts is chosen from a stored list and changed (cycling through a fixed list of orders) each session for a given participant. There are six videos shown in each of these concepts in total. Within each category, objects are assigned to comparison types (e.g. apple to table down vs. up) by choosing from a list of acceptable mappings, again incremented per session. There are six possible comparisons for the stay and fall events; three comparisons are assigned to stay and three to fall, with the selection again cycling through a random list of such assignments per session. Left/right placement, horizontal flipping of the left and right events, camera angles, and backgrounds are chosen randomly with the constraint that half of the ‘more probable’ events are on the left within each category. Calibration trials (grouped for purposes of assigning object intros with inertia videos) are placed early in the session at trials 3 and 6, so that they are always available for participants who completed enough trials for the session to be included.

This degree of specialized counterbalancing is, at present, best implemented via a single-purpose custom randomizer and specified directly in Javascript, as done here. However, ensuring that this protocol was possible motivated the availability of more general-purpose tools, such as a utility to retrieve data from past sessions for this child, which is now available in any randomizer frame – allowing researchers designing new randomizers not only to assign children to conditions based on stimuli they saw in past sessions in ways other than described here (e.g., assigning a child to a random condition he/she has not previously completed), but to alter study parameters based on children's previous *responses* (e.g., tuning difficulty of a task based on previous performance, or allowing parents to store preferences across study sessions).

### 5.6.3 Study 2: Infants' sensitivity to shape changes in triangles ("Baby Euclid")

**Overview.** This study was the first we implemented for another research group to run on Lookit, and constituted "round 1" of beta testing. The study protocol was specified by Moira Dillon and Elizabeth Spelke to replicate an in-lab finding using an alternation paradigm that infants are sensitive to changes in the shape of a triangle (Dillon, Izard, & Spelke, in review). In each preferential looking trial, infants are shown two streams of triangles; on one side the triangles alternate in size between a smaller and larger version, but on the other they additionally alternate in shape. The location of the "shape" stream alternates between left and right across four trials. Each trial begins with a brief calibration period so that coders can check that they are able to accurately detect the child's direction of gaze and that timing information is accurately recorded. See Figure 5.7 for a schematic of calibration and test trial sequences. The study protocol and analysis plan were preregistered; we plan to test 48 infants from 6.5 to 7.5 months. Data collection for this study is ongoing.

**Custom frame.** The 'exp-lookit-geometry-alternation' frame implements a single test trial of Study 2. Similar to 'exp-video-physics', it has three mutually exclusive states: intro, calibration, and test. During the intro, an attention-grabbing looping video is displayed while audio is played to update the parent on progress through the study and offer a chance to pause the study for a break. The frame proceeds to calibration once this audio has played, a webcam video connection has been established, and the intro duration has elapsed (generally rate-limiting). After calibration is completed, the test trial begins.

During the test trial, triangles are generated on-the-fly and displayed using svg (scalable vector graphics) tags in HTML. This allows randomly jittering the position, rotation, size, and reflection simply by changing these parameters in the svg tag (without changing the “base” triangle) for optimal clarity. Rotation is conducted around the triangle centroid because the base triangles are defined with the centroid at (0,0). Compared with showing image or video stimuli, this approach is also much faster (nothing to load), ensures clear images on any monitor size, and allows easy modification of parameters (e.g. triangle appearance or range of positions to jitter).

The study can only be paused and unpaused during the intro segment. If the frame is not displayed full-screen at the time the trial begins, the study is automatically paused and audio instructions instruct the parent to return to full-screen mode.

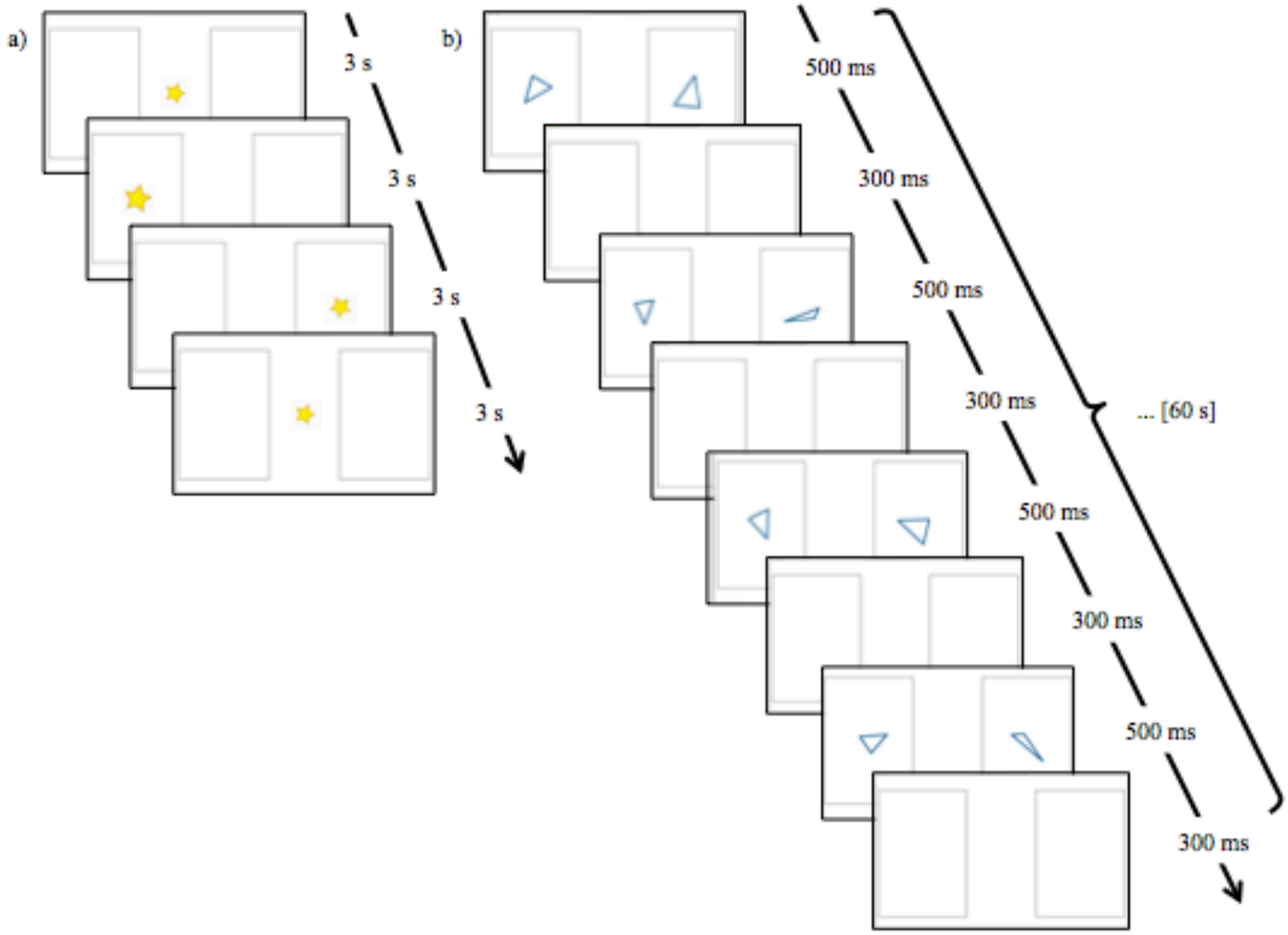



Figure 5.7: Screenshots from the frame developed for Study 2. a) Calibration segment: a small video (in this case a rotating star) is shown at center, left, right, and then center of the screen. b) Test segment: triangles generated on-the-fly are presented for 500 ms, with 300 ms pause between presentations.

In addition to the timing of each segment relative to the webcam video stream, this frame records event data about every triangle presentation since the stimuli vary randomly across participants and trials. Enough information is recorded to fully reproduce the sequence of triangle presentations a participant saw.

State changes are simplified from exp-video-physics by managing all segment transitions with a Ember observers, rather than directly starting one segment upon finishing the previous one; this better encapsulates the trial flow.

This frame takes as parameters:

- audio/video stimuli to use (intro video; calibration video and audio; music to play during trials; and audio to play during intro, when pausing/unpausing, when leaving fullscreen, and optionally at the end of the trial)

- timing information (length of intro, calibration, and test segments)
- triangle appearance (color and line width)
- counterbalancing information (whether to show the shape-alternating stream on the left, and whether to use as the common “context” shape a 45-60-75 or 15-45-120 triangle)

While still designed for a single experiment, this frame provides a more generalizable example of using several “states” within a frame. It also demonstrates the use of svg elements for generating and displaying clear stimuli on-the-fly.

**Condition assignment**. This study does not require condition assignment, but participants are counterbalanced in a 2x2 design: the position of the shape-changing stream may start on the left or right, and the “context” triangle may be one of two shapes.

Counterbalancing was achieved using a bespoke ‘geometry’ randomizer which creates a sequence of four trials, randomly selecting a starting position for the shape-changing stream (and then alternating) and randomly selecting a context triangle shape (which remains constant). To maximize researchers’ ability to make changes to the protocol simply by changing the study JSON document, without editing randomizer code, the geometry randomizer takes as arguments: the type of frame to use for trials, an object containing the frame parameters used in *every* frame this randomizer creates (i.e. the audio/video stimuli, timing information, and triangle appearance information shown above), and an object containing the possible values for the parameters to counterbalance (allowing researchers to, for instance, run only the one combination still needed for balanced conditions). This randomizer served as the starting point for the more general randomParameterSet randomizer described earlier, but also serves as a highly generalizable demonstration of how to generate a simple randomizer for a particular study. This may be desirable in cases where researchers are comfortable programming in Javascript and require a complex counterbalancing scheme and/or wish to describe logic more concisely. For instance, in the geometry study we specify the URL of the audio for each trial based on the trial number, rather than writing out the URL for each of four audio files, as shown below (this is the entire source code for the randomizer); if there were 50 short trials instead of 4 longer ones, this would be especially helpful.

```
var randomizer = function(frameId, frame) {
    var positionOptions = frame.counterbalance.startPositions;
    var contextOptions  = frame.counterbalance.contexts;
```

```
    var position = getRandomElement(positionOptions);
    var context  = getRandomElement(contextOptions);

    position = (position === 'left');
    context  = (context  === 'fat');

    var frames = [];
    var thisFrame = {};
    for (var iFrame = 0; iFrame < 4; iFrame++) { // make each of 4 trials
        thisFrame = {
            kind: frame.frameType,
            id: `${frameId}`,
            altOnLeft: position,
            context: context,
            audioSources: [ // audio announcement, URL by trial number
                {
                    'type': 'audio/mp3',
                    'src':
'https://s3.amazonaws.com/lookitcontents/geometry/mp3/video_0' + (iFrame + 1) +
'.mp3'
                },
                {
                    'type': 'audio/ogg',
                    'src':
'https://s3.amazonaws.com/lookitcontents/geometry/ogg/video_0' + (iFrame + 1) +
'.ogg'
                }
            ]
        };
        if (iFrame === 3) {
            thisFrame.endAudioSources = [
                {
                    'type': 'audio/mp3',
                    'src':
'https://s3.amazonaws.com/lookitcontents/geometry/mp3/all_done.mp3'
                },
                {
                    'type': 'audio/ogg',
                    'src':
'https://s3.amazonaws.com/lookitcontents/geometry/ogg/all_done.ogg'
                }
            ];
        }
        Object.assign(thisFrame, frame.frameOptions); // add common properties
        position = !position; // alternate position of changing stream
        frames.push(thisFrame); // add this frame to list
    }
    return [frames, {'position': position, 'context': context}];
};
export default randomizer;
```

### 5.6.4 Study 3: Facilitation of visual category formation by linguistic labels

**Overview.** The first study in "round 2" of beta testing was contributed by Bria Long and Mike Frank (Long & Frank, in prep). The custom frame to support this study has been implemented, and collaborators are currently working to write the JSON document to specify study behavior.

The aim of this study is to evaluate whether linguistic labels help infants represent visual categories, supporting better discrimination of objects in the newly created category from other perceptually similar objects. Infants are first familiarized with eight exemplars of a category, each displayed alone at the center of the screen, paired with either linguistic labels ("look, a DAX!") or non-linguistic tones ("beeeep, bip BEEP!"). Then they complete three silent test trials where preferential looking to novel exemplars of the familiarized category are paired with novel exemplars of a different category, followed by three word learning trials identical to the silent test trials but with either verbal prompts to find the familiar category ("Can you find the dax?") or non-linguistic tones. See Figure 5.8 for a schematic overview. Each trial begins with a brief central video to return the child's attention to the center of the screen; the trial starts once a specified amount of time has elapsed and the webcam connection is established. The parent can pause the study at any time; upon unpausing, they restart the current familiarization trial or skip to the next test trial. Upon leaving fullscreen mode, the study is automatically paused and audio is played prompting the parent to return to fullscreen mode. A calibration trial is included at the start of the study to check that researchers can detect looks to the left and right of the screen.

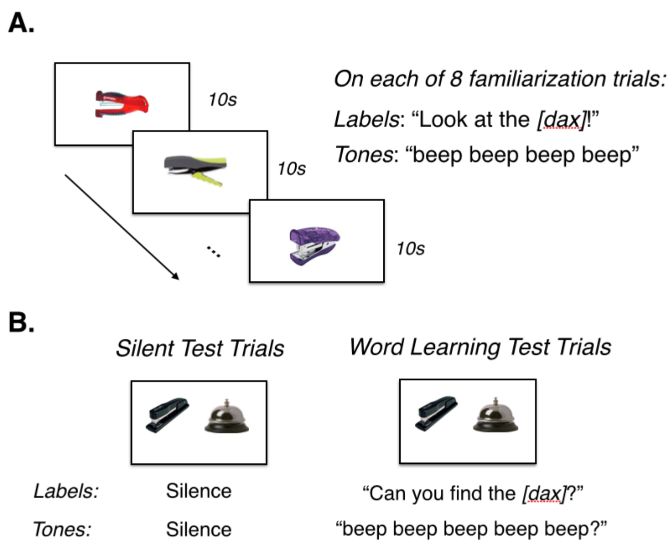



Figure 5.8. Schematic of protocol for Study 5, courtesy of Bria Long. (a) Example familiarization trials. (b) Example silent test trials and word learning trials.

**Custom frame.** The 'exp-lookit-preferential-looking' frame implements all trials (familiarization and test) for this study. The frame takes the following parameters:

- Stimuli: images to show at right, left, and/or center of screen; what audio to play at start of trial, when pausing/unpausing, when leaving fullscreen, and at end of trial (optional); attention-grabber video to use at start of trial; audio to play during calibration if showing a calibration trial.

- Base directory where stimuli are located (optional) and audio types to expect to find there (e.g. ['mp3', 'ogg']), to simplify specification of stimuli locations (researchers can simply provide filenames rather than full paths)
- Behavior: whether to allow pausing and restarting the trial; length of the test trial; how long to show the attention grabber video; whether this is a calibration trial and how long to make each calibration segment (rather than showing images, we simply show the calibration video at center, left, right, and center again)

This frame uses a state change logic similar to that in Study 2, moving from intro to test segments and between unpaused and paused states. The functionality of this frame is likely to be useful to other researchers directly or with very minimal edits as a way to conduct either looking time or preferential looking trials with static images.

**Condition assignment and counterbalancing.** Condition assignment and counterbalancing are implemented using the random-parameter-set randomizer discussed in Section 5.3, which was initially developed to support this study. Our collaborators were already prepared to provide a .csv sheet with the intended parameters for each subject number in order to support both assignment to the linguistic labels or tones condition and counterbalancing of specific object categories and stimulus order within categories. Below is an example "frameList" for the randomizer which demonstrates how the researchers could specify a study protocol with one of each type of frame needed:

```
"frameList": [
        { // Calibration trial
                "isCalibrationFrame": true,
                "calibrationLength": 3,
                "allowPausingDuringTest": false
            },
            { // Familiarization trial
                "centerImage": "FAMIMAGE",
                "testAudioSources": [
                    {
                        "stub": "TRAINAUDIO"
                    }
                ],
                "allowPausingDuringTest": true
            },
            { // Silent test trial
                "testAudioSources": [],
                "leftImage": "SILENTTESTLEFT",
                "allowPausingDuringTest": false,
                "rightImage": "SILENTTESTRIGHT"
            },
            { // Word learning trial
```

```
                    "testAudioSources": [
                        {
                            "stub": "TESTAUDIO"
                        }
                    ],
                    "endAudioSources": […],
                    "allowPausingDuringTest": true,
                    "leftImage": "TESTLEFT",
                    "rightImage": "TESTRIGHT"
                }
            ]
```

To randomly order stimuli within trials in a section, researchers can use a ‘permute’ randomizer as one of the trials in this frameList. This randomizer takes a list of frames and simply places them in a randomly chosen order.

### 5.6.5 Study 4: Preschoolers’ expectations of ingroup/outgroup moral obligations

**Overview.** The second study in “round 2” of beta testing was provided by Lisa Chalik and Yarrow Dunham. The necessary custom frame has been implemented, and our collaborators have provided all necessary stimuli and written and tested the JSON document to specify the study protocol. Testing is pending IRB approval.

This study focuses on how preschoolers (3-4 years old) view moral obligations between and across social groups; if the method can be successfully adapted for online testing, it will support broader extensions to real-world groups. Children are first introduced to two novel groups (flurps and zazzes). Then they complete a sequence of two-alternative forced-choice questions about the targets of positive and negative behaviors (e.g., if a flurp hits someone, does she hit a flurp or a zazz?) and about generalization of behaviors (e.g., if a flurp likes to play a particular game, does another flurp or a zazz also play that game?). Trials are self-paced; children respond verbally, and parents have the option to replay audio or prompt the child using language displayed on the screen during test trials. During both introductory and test audio, relevant images are highlighted as they are mentioned.

**Custom frame.** The ‘exp-lookit-story-page’ frame implements the necessary functionality for introductory and test trials (for example screenshots, see Figure 5.9). This frame supports a basic “storybook page” setup, with images placed on the screen within a display area and a sequence of audio files played. The researcher provides an arbitrary list of images and their desired positions on screen (coordinates and size), allowing for powerful customizability without complicating specification of a simple, single-image frame.

A list of audio files provided as a frame parameter is played in order, and a 'replay' button starts audio from the beginning. Audio specifications may optionally include a 'highlights' structure which specifies start and end times during the audio file to highlight images (indicated by IDs provided in the image specification), e.g. to put a border around a character who is currently "speaking" or referred to; in the example below, the image 'left' will be highlighted from seconds 3.02 – 5.60 and 4.38 – 7.74, while the image 'right' will be highlighted from seconds 5.75 to 8.89.   The free software package Audacity can be used to easily export start and end times of desired highlights using label tracks.

```
"highlights": [
        {"range": [3.02, 5.60], "image":     "left"},
        {"range": [5.75, 8.89], "image":     "right"},
        {"range": [4.38, 7.74], "image":     "left"},
]
```


Figure 5.9: Example screenshots from 'exp-lookit-story-page' frame for Study 3. a) Message displayed while establishing webcam connection if recording is enabled. b) Test trial showing three images – a protagonist (highlighted with green border) and two potential targets of her action, from the same and different social groups. c) Introductory trial showing images of two social groups, with the group currently mentioned (at right) highlighted.

If webcam recording is enabled, the video connection is established before displaying frame content. Researchers specify whether the parent presses 'next' to proceed (possible after completing audio) or the study should advance to the next frame automatically upon finishing the audio, and what text to display for parents. This frame is displayed fullscreen, but not paused or otherwise disabled if the user leaves fullscreen; a button simply appears prompting the user to return to fullscreen mode.

While designed specifically for this study, we expect that the frame itself is sufficiently general to be of broad use to researchers with minimal or no additional changes. This frame introduces the specification of stimulus (image/audio) via either a full path *or* a base directory and relative paths, allowing the use of shorter filenames and making it easier to switch out stimuli (by putting new versions in a different directory, and pointing the frame there in a single line of the JSON document).

**Condition assignment and counterbalancing.** This study requires a more typical degree of detailed counterbalancing than Study 2: half of the participants should see Flurps (vs. Zazzes) as protagonists; half of participants should see Flurps on the left (vs. right) during test trials; half of participants should hear stories about nice (vs. mean) actions taken by protatonists; and half of participants should start with questions about targets of actions (vs. generalization). Counterbalancing and condition assignment are achieved using the general-purpose random-parameter-set randomizer described in section 5.3. Researchers can either provide a list of possible pseudorandom trial orders, or randomize trial order within blocks by nesting a 'permute' randomizer within the frame list.

### 5.6.6 Study 4: Preschoolers' evaluations of politeness

**Overview.** The final study in "round 2" of beta testing was provided by Erica Yoon and Mike Frank. Testing is pending collaborators' availability; the necessary custom frame has been implemented, and our collaborators have provided stimuli and written and tested the JSON document to specify the study protocol.

This study evaluates preschoolers' judgments of more and less polite ways to express similar sentiments (e.g., ordering versus requesting that someone turn out the lights). During each test trial, a protagonist is introduced; one character appears and speaks; a second character appears and speaks; the participant can review what either character said by clicking; and the participant answers two verbal questions by clicking on one of the two characters.

**Custom frame.** The 'exp-lookit-dialogue-page' frame implements the functionality needed for this study. Each frame represents a single phase of a test trial – introduction, a character appearing and speaking, or a verbal question. This frame is essentially an extension of exp-lookit-story-page from Study 4, with an identical layout but additional audio functionality. First, images appear and any main narration audio is played. Next, the user can click on images to play additional audio associated with the image, or (for a choice trial) click one of the images to select it as an answer. A trial with only main narration audio can also simply auto-proceed when audio is finished.

Researchers specify an arbitrary list of images, including the image source, position, any animation to show at the start of the trial (fade in, fade out, fly from left, fly from right), audio associated with this image, whether audio must be played before moving on from this frame, and any text to display above the image. This allows specification of each phase of the test trial for this study as an exp-lookit-dialogue-page frame (see Figure 5.10).

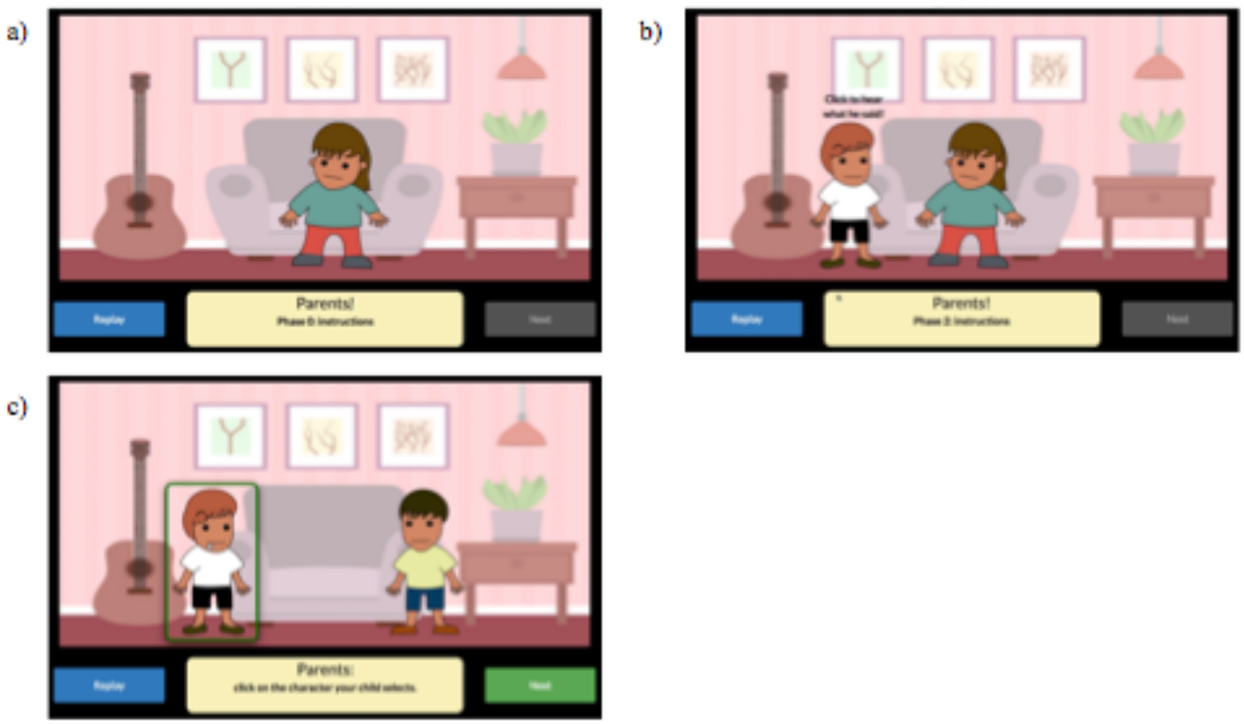



Figure 5.10: Screenshots of exp-lookit-dialogue-page frames. a) Initial phase introducing protagonist. b) First speaker enters and user clicks to hear what he says. c) Choice phase where child selects one of the two speakers.

Although, again, this frame was written specifically for one study, designing it to cover each phase of the test trial encouraged sufficient generalization for it to be useful with minimal or no modification for future storybook-style studies.

**Condition assignment and counterbalancing.** Children are assigned to one of several sequences of trials, with speaker position, voiceovers, and item order counterbalanced.

Counterbalancing is handled via the general-purpose random-parameter-set randomizer, further demonstrating its versatility. Furthermore, to condense and structure the trial list, randomizers are nested, with a single 'inner' randomizer specifying the sequence of six phases within each test trial, as demonstrated in Section 5.3.

## 5.7 Video coding and data analysis

Currently, video processing and coding workflow fall outside the scope of the Lookit and Experimenter platforms. That is, researchers download systematically-named video files and study data in CSV or JSON format, and are responsible for setting up to analyze those data on their own. However, we have developed tools for our own workflow in processing data from Studies 1 and 2 that may serve as a starting point for other researchers with similar needs or to inform future official integration with Experimenter.

The python script coding.py, freely available in the "coding-workflow" branch of our fork of the lookit codebase at https://github.com/kimberscott/lookit/tree/coding-workflow/scripts, provides basic functionality for video organization, processing, and coding. It includes a command-line interface for retrieving updated video and data from a study, generating human-readable spreadsheets with summaries of study data, and saving information from an arbitrary number of individual coders.

Data collected on Lookit are initially stored in databases on the Open Science Framework (OSF). To see and process data in the lab, we retrieve updated data from the OSF. Locally, we store *session* and *account*, in addition to *coding* data generated by automatic video processing and human coding, in .bin files readable by python. To interact with the data, a coder "fetches" a temporary .csv file to view and edit in Excel (by running `python coding.py fetchcodesheet --study STUDYNAME --coder CODERNAME`). The coder might record information such as whether he/she has coded each session or what a child's verbal response was. Information about which other coders have also coded this session is available on the coding spreadsheet to allow coders to coordinate efforts. After making and saving changes to the csv file, the coder then "commits" the changes, which edits coder-specific fields in the .bin file with the primary copy of coding data for this study. (The coder cannot edit other coders' fields or remove or edit any other data by committing changes, and the previous version of the .bin file is backed up automatically.)

Coding spreadsheets only show sessions for which consent has already been confirmed, in order to prevent accidental viewing of videos where we cannot confirm that the parent provided informed consent to participate. A researcher must first fetch a consent spreadsheet, which shows all sessions; record whether consent was confirmed for a given session; and commit the sheet for confirmed sessions to appear in the code sheet.

Study parameters specified in coding_settings.py per study allow researchers to customize the appearance of spreadsheets for their studies, as well as the video processing that occurs when downloading new study videos. Researchers can specify the current coder names to expect; usernames of test accounts to ignore; field names to include or exclude in coding spreadsheets; how many videos are expected in a complete session and which frames are expected to record video; coder-specific columns to add to coding spreadsheets; whether/how to trim study videos (position relative to start or end of the video to begin at, or an event name to begin at); and whether to create concatenated study videos (combining all trimmed video from a given session) before consent is confirmed. When running an update, any new videos are downloaded and organized, with consent videos separated into a single directory per study and remaining videos placed in directories per session; details of the videos (e.g. duration) are stored with coding data.

Currently, coding.py also allows sending feedback entered on a consent sheet to the session database on OSF so that parents can view session feedback, and another ad-hoc script reminder_emails.py demonstrates sending participants reminder emails for a longitudinal study. Due to backend changes, these scripts will no longer be supposed as written, but the Experimenter API will still support feedback and email functionality.

## 5.8 Initial data collection

Data collection is currently in progress for Studies 1 and 2, and pilot data were additionally collected for Study 1. We can successfully record high-quality webcam video and capture session data on the new platform. Figure 5.11 shows sample webcam video frames.


Figure 5.11: Example frames from webcam video recorded on the new platform during piloting for Study 1. Top panel: parents and their infants during consent statements. Bottom panel: infants looking to the left and right of the monitor.

During piloting of Study 1, intended to check that data collection was working and children could complete the task, we collected valid consent recordings for 107 sessions from 93 unique participants between 4 and 13 months of age, primarily recruited via Amazon Mechanical Turk. An additional 36 consent recordings were invalid due to lack of audio (n = 13), failure to read the consent statement (n = 18), or other technical problems (n = 5). Of these sessions, 43 were included in analysis; sessions were excluded if the participant completed fewer than half of the test trials (n = 41), video data was withdrawn (n = 5), children were out of the age range or no child was present (n = 5), video could not be coded due to blurriness, poor lighting or camera angle, low framerate, or delays in starting recording (n = 10), or the child was generally out of frame or not looking (n = 3). Of the 43 included participants, 28 were female; the average age was 8.6 months (range 4.0 – 12.8 months); and 37 were born at full term (37 weeks or more).

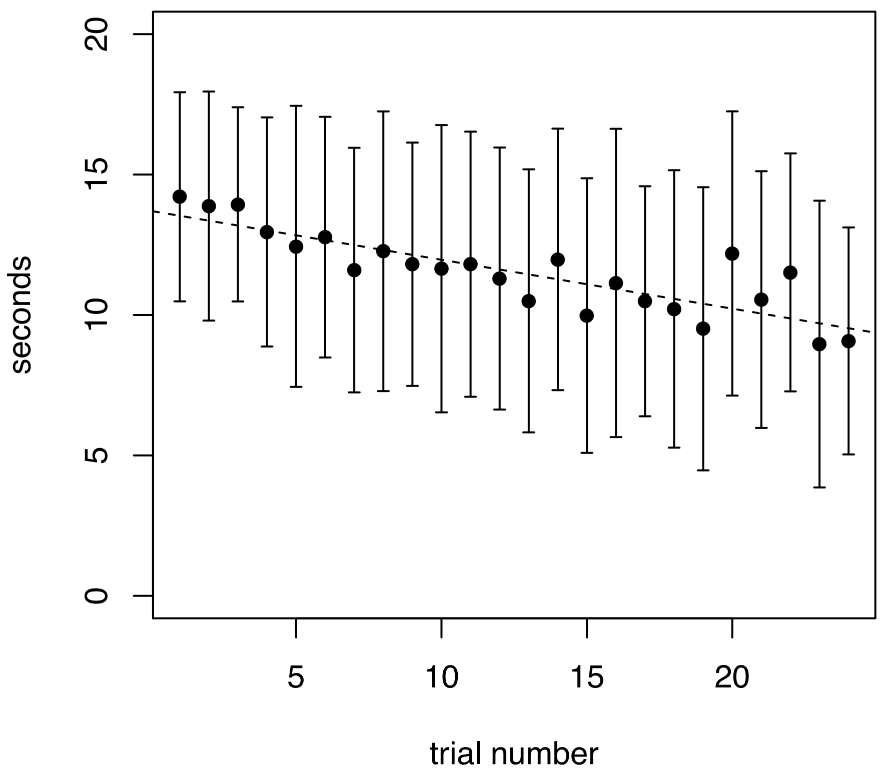



Figure 5.12: Mean total looking time per trial number across Study 1 pilot data (43 participants), excluding trials with total looking times under 2 s or after participants left the study. Total trial duration was 20 s. Error bars show standard deviations across participants.

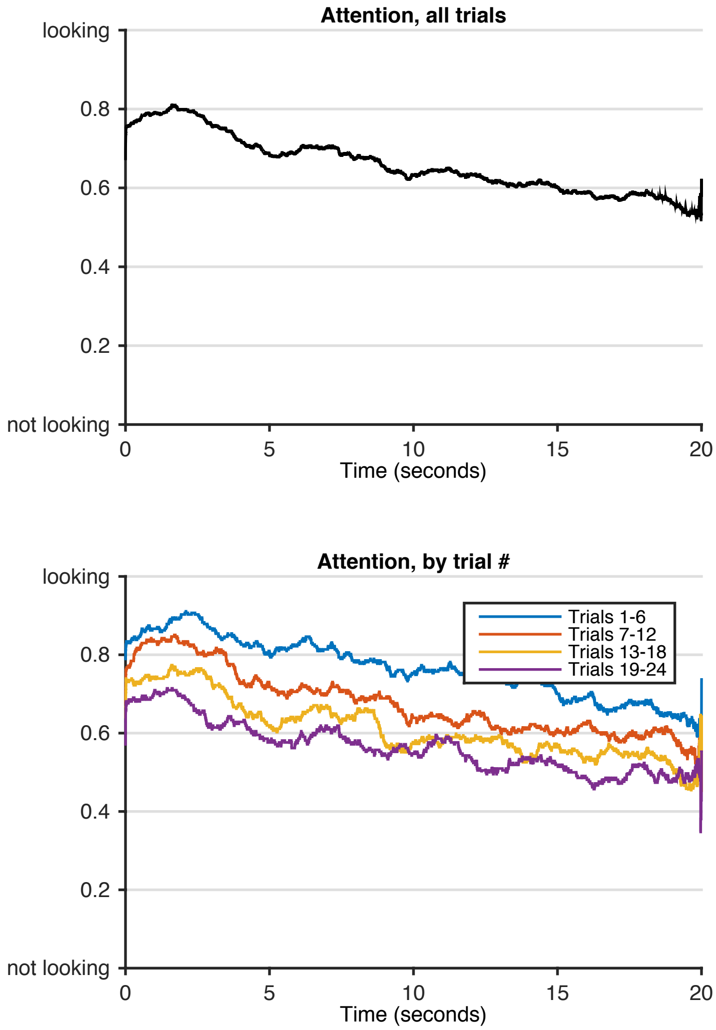



Figure 5.13. Proportion of children looking to the screen over the course of Study 1 pilot trials, overall (top panel) and separated by trial number (bottom panel). (Noise near 20s is due to slight variation in video durations, such that some videos are very slightly shorter.)

Pilot data showed that children's total looking time declined over the course of 24 trials (Figure 5.12) and over the course of individual trials (Figure 5.13), as expected, but remained at approximately 50% even at the end of the study. Looks to the left and right were clearly observed during calibration (Figure 5.14) and preferences were fairly stable across the course of the trial (Figure 5.15). Children showed clear preferences for the "more salient" videos during salience controls, as shown in Figure 5.16.

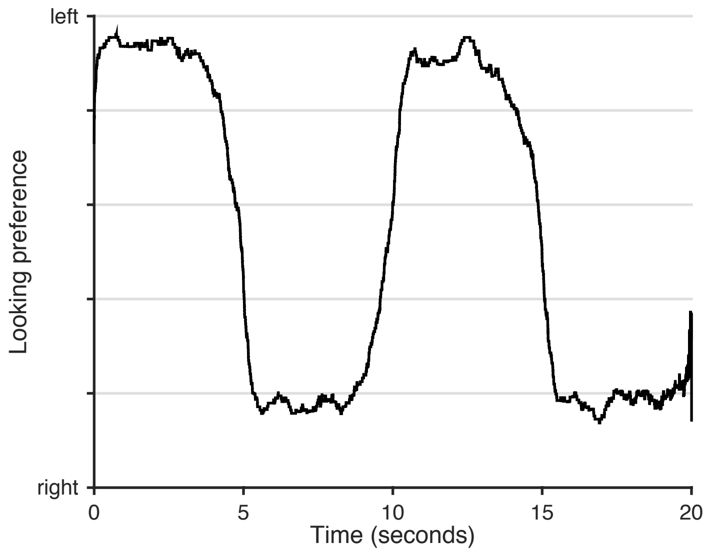



Figure 5.14: Average looking preference (# children looking left / # children looking at screen, per ms) across 192 calibration trials from 43 participants recorded during piloting of Study 1.

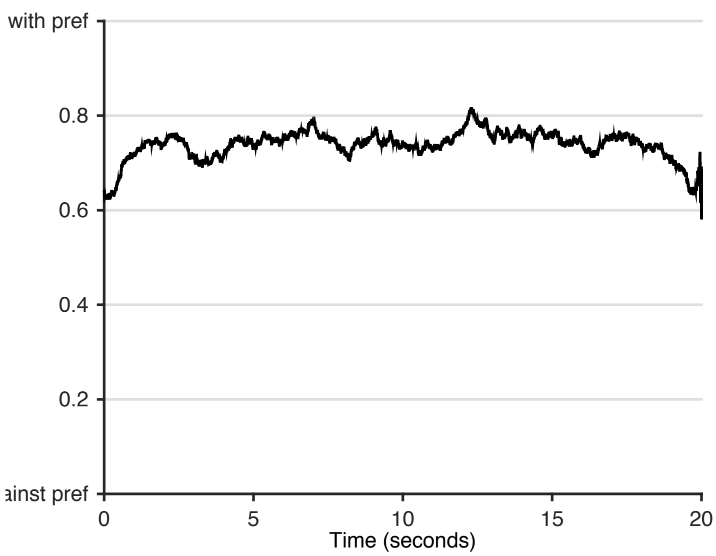



Figure 5.15: Average looking preference in the direction of overall trial preference (# children looking in the same direction as their overall preference this trial / # children looking at screen, per ms) across 740 non-calibration trials from 43 participants in pilot of Study 1.

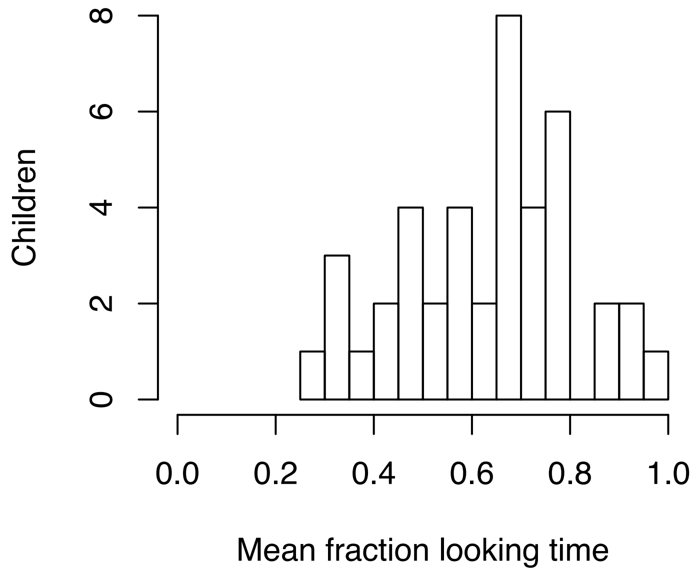



Figure 5.16. Histogram of participants' mean preferences for the more salient event (looking time to more salient event / total looking time) in salience trials during piloting of Study 1.

We did not expect pilot data to provide useful information with regard to the magnitude and direction of group-level preferences when viewing the test events, where we expect substantially weaker preferences and potentially more variation. However, we do observe varying preferences across the comparisons used in the pilot study. We specifically focused on comparisons where effects have been reported using very similar stimuli, and included only trials from children of at least the age where the effect was reported. Using stimuli similar to our "ramp" contrasts, Kim & Spelke (1992, Experiment 3) report preferences for the unnatural movement at 7 months but not 5 months. Baillargeon reports preferences for unnatural support events similar to our "stay" events, with sensitivity to contact vs. no contact arising 3 months (analogous to our mostly-on vs. near comparison), followed by contact on the top vs. side of the supporting surface at 4.5 months (analogous to our mostly-on vs. next-to comparison) and sensitivity to the amount of contact at 6.5 months (analogous to our mostly-on vs. slightly-on comparison); for review see Baillargeon (1995). In contrast, Friedman (2002) reports preferences for the *natural* versions of events similar to our "toss" events: 8-month-olds looked longer at a ball moving up from one hand to another than at it falling down. (Although this work investigated sensitivity to temporal reversals rather than physics per se, many of the reversals could be construed to violate either gravity or directionality of time, and infants demonstrated consistent preferences for the familiar versions of events.) In all of the prior work described, looking time was measured to single events rather than simultaneously-presented pairs.

The mean preferences per event in the pilot data starting at the relevant reported ages are summarized in Table 5.2. We observe a potential preference for the natural version of the "toss" events and for the unnatural slightly-on rather than mostly-on "stay" events. However, we do not observe consistent preferences for the unnatural versions of the "stay" events across comparisons, nor do we see a clear preference in the case of the "ramp" events.

| Comparison | Minimum age | Direction of preference from literature | Mean preference for unnatural version, Lookit pilot data (N); generous inclusion | Mean preference for unnatural version, Lookit pilot data (N); stringent inclusion |
|---|---|---|---|---|
| Ramp, down:up | 6.5 months | Unnatural | 0.48 (27) | 0.52 (22) |
| Toss, down:up | 7.5 months | Natural | 0.40 (24) | 0.45 (20) |
| Stay, mostly-on:near | 3 months | Unnatural | 0.37 (39) | 0.37 (14) |
| Stay, mostly-on:next-to | 4.5 months | Unnatural | 0.48 (33) | 0.65 (14) |
| Stay, mostly-on:slightly-on | 6.5 months | Unnatural | 0.55 (23) | 0.60 (12) |

Table 5.2. Mean preferences across children for the unnatural versions of physical events in pilot data of Study 1. Preferences are computed per trial as the fraction of time spent looking to the unnatural version divided by total looking time; the preference per child is the mean of all included trials of this type. "Generous" inclusion criteria include all trials with a total looking time of at least 2 s from all participants whose data could be coded. "Stringent" inclusion criteria include all trials with a total looking time of at least 10 s from all participants where the mean calibration metric (fraction of time spent looking at the expected side during calibration trials, where a ball moves back and forth on the screen) across all calibration trials was at least 0.75. For each comparison type, only children of at least the minimum age are included.

Following piloting of Study 1, we have begun to collect data for the longitudinal study; we plan to collect at least 12 usable sessions from at least 50 participants. To date, we have collected 148 usable sessions from 27 unique participants. We have at least 12 usable sessions from 5 participants, and at least 6 usable sessions from another 5 participants. Analysis of data from this study is pending preregistration of the analysis plan.

Data collection for Study 2 is also underway; to date, 16 unique participants have completed the study, yielding 12 usable sessions, and preliminary coding shows intercoder correlation of 0.96 for total looking time per trial (Moira Dillon, personal communication, July 23 2017).

The most common technical challenge in collecting webcam video has been occasional missing audio streams, which necessitate contacting parents to confirm consent via email even if audio is not required to use data from the rest of the study. Audio was initially missing in videos from approximately 18% of participants, with problems consistent to specific participants. All participants contacted about this problem reported using Chrome on a Mac to complete the study, and being able to detect webcam audio on a test page using the same browser; work to detect and minimize instances of missing audio is ongoing.

## 5.9 Conclusion

The major new developments in the reengineered Lookit platform are threefold:

- The Experimenter platform provides a system for flexible, powerful specification of study protocols without any programming required, by using Ember components to implement modular study "frames." Advanced users have the option to provide an extended codebase with their own custom frames and/or randomizers.
- Code for each study is siloed to prevent unintended hanges to existing studies upon updates. Researchers can preview and test their studies directly via the Experimenter interface.
- Granular permissions are available to control researchers' access to and control over studies. Individual researchers can be granted access to particular studies, but by default can only view, edit, start/stop, or download data from studies they created.

Substantial improvements have also been made to video recording, security, and databases for session and account data. We have tested data collection and study development by implementing five studies on the new platform, four in collaboration with other labs and following their precise protocol requirements.

While Lookit is by no means now "done" – there are many features that would help both researchers and participants use and engage with the platform, described in Chapter 6 – it is well-positioned to begin supporting more independent research by other developmental researchers. We look forward to creative uses of this method. Although it will not be appropriate for every study, online research can expand access to both more representative populations and rare populations, and make it easier to conduct large-scale longitudinal studies, detect small and graded effects, generate data sufficient for testing computational models, and assess individual differences and developmental change. We hope this tool will also be used to replicate classic effects and make the replication of new results easier.  Finally, in connecting families and scientists, Lookit offers exciting new opportunities for education and outreach. As a venue for "citizen science" as well as scientific research, our goal for Lookit is to expand the scope of both the questions we ask and the people we reach.

# References


Alvarez, G. A., & Cavanagh, P. (2004). Independent attention resources for the left and right visual hemifields. *Journal of Vision*, *4*(8), 29–29. http://doi.org/10.1167/4.8.29

Amazon. (2014). Amazon Mechanical Turk Participation Agreement. Retrieved from https://requester.mturk.com/policies/conditionsofuse

Antell, S. E. G., & Caron, A. J. (1985). Neonatal perception of spatial relationships. *Infant Behavior and Development*, *8*(1), 15–23. http://doi.org/10.1016/S0163-6383(85)80013-8

Aslin, R. N. (2007). What's in a look? *Developmental Science*, *10*(1), 48–53. http://doi.org/10.1111/j.1467-7687.2007.00563.x

Baillargeon, R. (1995). *Physical reasoning in infancy*. (M. S. Gazzaniga, Ed.)*The Cognitive Neurosciences*. Cambridge, MA: MIT Press.

Baillargeon, R., Spelke, E. S., & Wasserman, S. (1985). Object permanence in five-month-old infants. *Cognition*, *20*(3), 191–208. http://doi.org/10.1016/0010-0277(85)90008-3

Bakker, M., van Dijk, A., & Wicherts, J. M. (2012). The Rules of the Game Called Psychological Science. *Perspectives on Psychological Science*, *7*(6), 543–554. http://doi.org/10.1177/1745691612459060

Bornstein, M. H., Hahn, C., Suwalsky, J.T.D., & Haynes, O. M. (2003). Socioeconomic status, parenting, and child development: The Hollingshead Four-Factor Index of Social Status and the socionomic index of occupations. In M. H. Bornstein & R. H. Bradley (Eds.), *Socioeconomic status, parenting, and child devleopment* (pp. 29–82). Mahwah, NJ: Lawrence Erlbaum Associates.

Braginsky, M., Yurovsky, D., Marchman, V. A., & Frank, M. C. (2017). Consistency and variability in word learning across languages.

Brainard, D. H. (1997). The Psychophysics Toolbox. *Spatial Vision*, *10*, 433–436. http://doi.org/10.1163/156856897X00357

Buhrmester, M., Kwang, T., & Gosling, S. D. (2011). Amazon's Mechanical Turk: A New Source of Inexpensive, Yet High-Quality, Data? *Perspectives on Psychological Science*, *6*(1), 3–5. http://doi.org/10.1177/1745691610393980

Burr, D., & Ross, J. (2008). A Visual Sense of Number. *Current Biology*, *18*, 425–428. http://doi.org/10.1016/j.cub.2008.02.052

Csibra, G., & Gergely, G. (2009). Natural pedagogy. *Trends in Cognitive Sciences*, *13*(4), 148–153. http://doi.org/10.1016/j.tics.2009.01.005

Csibra, G., Hernik, M., Mascaro, O., Tatone, D., & Lengyel, M. (2016). Statistical treatment of looking-time data. *Developmental Psychology*, *52*(4), 521–536. http://doi.org/10.1037/dev0000083

Dannemiller, J. L., & Hanko, S. A. (1987). A test of color constancy in 4-month-old human infants. *Journal of Experimental Child Psychology*, *44*(2), 255–267. http://doi.org/10.1016/0022-0965(87)90033-6

de Leeuw, J. R. (2015). jsPsych: A JavaScript library for creating behavioral experiments in a Web browser. *Behavior Research Methods*, *47*(1), 1–12. http://doi.org/10.3758/s13428-014-0458-y

de Schonen, S., & Bry, I. (1987). Interhemispheric communication of visual learning: a developmental study in 3-6-month old infants. *Neuropsychologia*, *25*(4), 601–12.

Deruelle, C., & de Schonen, S. (1991). Hemispheric asymmetries in visual pattern processing in infancy. *Brain and Cognition*, *16*(2), 151–179. http://doi.org/10.1016/0278-2626(91)90004-R

Elberger, A. J. (1984). The existence of a separate, brief critical period for the corpus callosum to affect visual development. *Behavioural Brain Research*, *11*(3), 223–231. http://doi.org/10.1016/0166-4328(84)90214-6

Feigenson, L., Carey, S., & Spelke, E. (2002). Infants ' Discrimination of Number vs. Continuous Extent. *Cognitive Psychology*, *44*, 33–66. http://doi.org/10.1006/cogp.2001.0760

Feigenson, L., Dehaene, S., & Spelke, E. (2004). Core systems of number. *Trends Cogn Sci*, *8*(7), 307–314. http://doi.org/10.1016/j.tics.2004.05.002

Ferhat, O., & Vilariño, F. (2016). Low Cost Eye Tracking : The Current Panorama. *Computational Intelligence and Neuroscience*, *2016*. http://doi.org/10.1155/2016/8680541

Frank, M. C., Bergelson, E., Bergmann, C., Cristia, A., Floccia, C., Gervain, J., … Yurovsky, D. (2017). A collaborative approach to infant research: Promoting reproducibility, best practices, and theory-building. *Infancy*, *22*(4), 421–435. http://doi.org/10.1111/infa.12182

Franklin, A., Drivonikou, G. V, Bevis, L., Davies, I. R. L., Kay, P., & Regier, T. (2008). Categorical perception of color is lateralized to the right hemisphere in infants, but to the left hemisphere in adults. *Proceedings of the National Academy of Sciences of the United States of America*, *105*(9), 3221–5. http://doi.org/10.1073/pnas.0712286105

Friedman, W. J. (2002). Arrows of time in infancy: the representation of temporal-causal invariances. *Cognitive Psychology*, *44*(3), 252–96. http://doi.org/10.1006/cogp.2001.0768

Gazzaniga, M. S. (2005). Forty-five years of split-brain research and still going strong. *Nature Reviews. Neuroscience*, *6*(8), 653–9. http://doi.org/10.1038/nrn1723

Giedd, J. N., Blumenthal, J., Jeffries, N. O., Castellanos, F. X., Liu, H., Zijdenbos, A., … Rapoport, J. L. (1999). Brain development during childhood and adolescence: a longitudinal MRI study. *Nature Neuroscience*, *2*(10), 861–863. http://doi.org/10.1038/13158

Gliga, T., & Dehaene-Lambertz, G. (2007). Development of a view-invariant representation of the human head. *Cognition*, *102*(2), 261–88. http://doi.org/10.1016/j.cognition.2006.01.004

Gouteux, S., & Spelke, E. S. (2001). Children's use of geometry and landmarks to reorient in an open space. *Cognition*, *81*(2), 119–148. http://doi.org/10.1016/S0010-0277(01)00128-7

Granrud, C. E. (2006). Size constancy in infants: 4-month-olds' responses to physical versus retinal image size. *Journal of Experimental Psychology. Human Perception and Performance*, *32*(6), 1398–1404.

Gweon, H., Tenenbaum, J. B., & Schulz, L. E. (2010). Infants consider both the sample and the sampling process in inductive generalization. *Proceedings of the National Academy of Sciences of the United States of America*, *107*(20), 9066–71. http://doi.org/10.1073/pnas.1003095107

Hagedorn, J., Hailpern, J., & Karahalios, K. (2008). VCode and VData: illustrating a new framework for supporting the video annotation workflow. *Proceedings of the Workshop on Advanced Visual Interfaces AVI*, 317–321.

http://doi.org/http://doi.acm.org/10.1145/1385569.1385622

Halberda, J., Sires, S. F., & Feigenson, L. (2006). Multiple spatially-overlapping sets can be enumerated in parallel. *Psychological Science*, *17*(7), 572–576. http://doi.org/10.1111/j.1467-9280.2006.01746.x

Hermer, L., & Spelke, E. (1996). Modularity and development: The case of spatial reorientation. *Cognition*, *61*(3), 195–232. http://doi.org/10.1016/S0010-0277(96)00714-7

Hoffman, L. W. (2003). Methodological issues in studies of SES, parenting, and child development. In M. H. Bornstein & R. H. Bradley (Eds.), *Socioeconomic status, parenting, and child devleopment* (pp. 125–143). Mahwah, NJ: Lawrence Erlbaum Associates.

Houston-Price, C., & Nakai, S. (2004). Distinguishing novelty and familiarity effects in infant preference procedures. *Infant and Child Development*, *13*(4), 341–348. http://doi.org/10.1002/icd.364

Hunter, M. A., Ames, E. W., & Koopman, R. (1983). Effects of stimulus complexity and familiarization time on infant preferences for novel and familiar stimuli. *Developmental Psychology*, *19*(3), 338–352. http://doi.org/10.1037/0012-1649.19.3.338

Ioannidis, J. P. A., Munafo, M. R., Fusar-Poli, P., Nosek, B. A., & David, S. P. (2014). Publication and other reporting biases in cognitive sciences : detection , prevalence , and prevention. *Trends in Cognitive Sciences*, *18*(5). http://doi.org/10.1016/j.tics.2014.02.010

John, L. K., Loewenstein, G., & Prelec, D. (2012). Measuring the Prevalence of Questionable Research Practices With Incentives for Truth Telling. *Psychological Science*. http://doi.org/10.1177/0956797611430953

Kidd, C., Piantadosi, S. T., & Aslin, R. N. (2012). The Goldilocks effect: human infants allocate attention to visual sequences that are neither too simple nor too complex. *PloS One*, *7*(5), e36399. http://doi.org/10.1371/journal.pone.0036399

Kim, I. K., & Spelke, E. S. (1992). Infants' Sensitivity to Effects of Gravity on Visible Object Motion. *Journal of Experimental Psychology: Human Perception and Performance*, *18*(2), 385–393. http://doi.org/10.1037/0096-1523.18.2.385

Klahr, D., & Dunbar, K. (1988). Dual space search during scientific reasoning. *Cognitive Science*.

Lakens, D., & Evers, E. R. K. (2014). Sailing From the Seas of Chaos Into the Corridor of Stability. *Perspectives on Psychological Science*, *9*(3), 278–292. http://doi.org/10.1177/1745691614528520

Le Grand, R., Mondloch, C. J., Maurer, D., & Brent, H. P. (2003). Expert face processing requires visual input to the right hemisphere during infancy. *Nature Neuroscience*, *6*(10), 1108–1112. http://doi.org/10.1038/nn1121

Leopold, D. A., O'Toole, A. J., Vetter, T., & Blanz, V. (2001). Prototype-referenced shape encoding revealed by high-level aftereffects. *Nature Neuroscience*, *4*(1), 89–94. http://doi.org/10.1038/82947

Liégeois, F., Bentejac, L., & De Schonen, S. (2000). When does inter-hemispheric integration of visual events emerge in infancy? A developmental study on 19- to 28-month-old infants. *Neuropsychologia*, *38*(10), 1382–1389. http://doi.org/10.1016/S0028-3932(00)00041-5

Long, B., & Frank, M. C. (n.d.). *What do infants encode when they learn new visual categories?*
MacLin, O. H., Nelson, H. A., & Webster, M. A. (1996). Figural after-effects in the perception of faces. *Investigative Ophthalmology and Visual Science*, *37*(3), 647–653. http://doi.org/10.3758/BF03212974
Malinowski, P., Fuchs, S., & Matthias, M. M. (2007). Sustained division of spatial attention to multiple locations within one hemifield. *Neuroscience Letters*, *414*, 65–70. http://doi.org/10.1016/j.neulet.2006.12.001
Mathôt, S., Schreij, D., & Theeuwes, J. (2012). OpenSesame: An open-source, graphical experiment builder for the social sciences. *Behavior Research Methods*, *44*(2), 314–324. http://doi.org/10.3758/s13428-011-0168-7
McCrink, K., & Wynn, K. (2004). Large-number addition and subtraction by 9-month-old infants. *Psychological Science*, *15*(11), 776–81. http://doi.org/10.1111/j.0956-7976.2004.00755.x
McKenzie, B. E., Tootell, H. E., & Day, R. H. (1980). Development of visual size constancy during the 1st year of human infancy. *Developmental Psychology*, *16*(3), 163–174. http://doi.org/10.1037/0012-1649.16.3.163
Mcshane, B. B., & Böckenholt, U. (2016). You Cannot Step Into the Same River Twice : When Power Analyses Are Optimistic. http://doi.org/10.1177/1745691614548513
Milewski, A. E. (1979). Visual discrimination and detection of configurational invariance in 3-month infants. *Developmental Psychology*, *15*(4), 357–363.
Nosek, B. A., & Lakens, D. (2014). Registered reports: A method to increase the credibility of published results. *Social Psychology*, *45*(3), 137–141. http://doi.org/10.1027/1864-9335/a000192
Onishi, K. H., & Baillargeon, R. (2005). Do 15-month-old Infants Understand False Beliefs? *Science (New York, N.Y.)*, *308*(5719), 255–8. http://doi.org/10.1126/science.1107621
Open Science Collaboration. (2015). Estimating the reproducibility of psychological science. *Science*, *349*(6251), aac4716-aac4716. http://doi.org/10.1126/science.aac4716
Ott, M. (2013). Unique Turker. Retrieved from https://uniqueturker.myleott.com/
Paolacci, G., Chandler, J., & Ipeirotis, P. (2010). Running experiments on amazon mechanical turk. *Judgment and Decision Making*, *5*(5), 411–419. http://doi.org/10.2139/ssrn.1626226
Pashler, H., & Wagenmakers, E. J. (2012). Editors' introduction to the special section on replicability in psychological science: A crisis of confidence? *Perspectives on Psychological Science*, *7*(6), 528–530. http://doi.org/10.1177/1745691612465253
Pasquini, E. S., Corriveau, K. H., Koenig, M. A., & Harris, P. L. (2007). Preschoolers monitor the relative accuracy of informants. *Developmental Psychology*, *43*(5), 1216–1226. http://doi.org/10.1037/0012-1649.43.5.1216
Peirce, J. W. (2007). PsychoPy-Psychophysics software in Python. *Journal of Neuroscience Methods*, *162*(162), 8–13. http://doi.org/10.1016/j.jneumeth.2006.11.017
Piantadosi, S. T., Kidd, C., & Aslin, R. (2014). Rich analysis and rational models: Inferring individual behavior from infant looking data. *Developmental Science*, *17*(3), 321–337. http://doi.org/10.1111/desc.12083
R Core Team. (2015). R: A language and environment for statistical computing. Retrieved from http://www.r-project.org/
Rand, D. G. (2012). The promise of Mechanical Turk: how online labor markets can help

theorists run behavioral experiments. *Journal of Theoretical Biology*, *299*, 172–9. http://doi.org/10.1016/j.jtbi.2011.03.004

Richards, J. E. (1997). Effects of attention on infants' preference for briefly exposed visual stimuli in the paired-comparison recognition-memory paradigm. *Developmental Psychology*, *33*(1), 22–31. http://doi.org/10.1037/0012-1649.33.1.22

Roder, B. J., Bushneil, E. W., & Sasseville, A. M. (2000). Infants' Preferences for Familiarity and Novelty During the Course of Visual Processing. *Infancy*, *1*(4), 491–507. http://doi.org/10.1207/S15327078IN0104_9

Rose, S. A., Gottfried, A. W., Melloy-Carminar, P., Bridger, W. H., Mello-Carmina, P., & Bridger, W. H. (1982). Familiarity and novelty preferences in infant recognition memory: Implications for information processing. *Developmental Psychology*, *18*(5), 704–713. http://doi.org/10.1037/0012-1649.18.5.704

Roser, M. E., Fugelsang, J. A., Dunbar, K. N., Corballis, P. M., & Gazzaniga, M. S. (2005). Dissociating processes supporting causal perception and causal inference in the brain. *Neuropsychology*, *19*(5), 591–602. http://doi.org/10.1037/0894-4105.19.5.591

Ross, J., & Burr, D. C. (2010). Vision senses number directly. *Journal of Vision*, *10*(2), 10.1-8. http://doi.org/10.1167/10.2.10

Salamy, A. (1978). Commissural transmission: maturational changes in humans. *Science*, *200*(4348), 1409–11. http://doi.org/10.1126/science.208144

Scott, K., Chu, J., & Schulz, L. (2017). Lookit (Part 2): Assessing the viability of online developmental research, results from three case studies. *Open Mind: Discoveries in Cognitive Science*, *1*(1), 15–29. http://doi.org/10.1162/opmi_a_00001

Scott, K. M., & Schulz, L. E. (2014). Interhemispheric integration of visual concepts in infancy. *Proceedings of the Annual Meeting of the Cognitive Science Society*, *36*, 1395–1400.

Scott, K., & Schulz, L. (2017). Lookit (Part 1): A new online platform for developmental research. *Open Mind: Discoveries in Cognitive Science*, *1*(1), 1–14. http://doi.org/10.1162/opmi_a_00002

Sereno, A. B., & Kosslyn, S. M. (1991). Discrimination within and between hemifields: A new constraint on theories of attention. *Neuropsychologia*, *29*(7), 659–675. http://doi.org/10.1016/0028-3932(91)90100-M

Shapiro, D. N., Chandler, J., & Mueller, P. a. (2013). Using Mechanical Turk to study clinical populations. *Clinical Psychological Science*, *1*(January), 213–220. http://doi.org/10.1177/2167702612469015

Simmons, J. P., Nelson, L. D., & Simonsohn, U. (2011). False-Positive Psychology: Undisclosed Flexibility in Data Collection and Analysis Allows Presenting Anything as Significant. *Psychological Science*, *22*(11), 1359–1366. http://doi.org/10.1177/0956797611417632

Slater, A., Mattock, A., & Brown, E. (1990). Size constancy at birth: Newborn infants' responses to retinal and real size. *Journal of Experimental Child Psychology*, *49*(2), 314–322. http://doi.org/10.1016/0022-0965(90)90061-C

Slaughter, V., & Suddendorf, T. (2007). Participant loss due to “fussiness” in infant visual paradigms: A review of the last 20 years. *Infant Behavior and Development*, *30*, 505–514. http://doi.org/10.1016/j.infbeh.2006.12.006

Smith, L., & Yu, C. (2008). Infants rapidly learn word-referent mappings via cross-situational statistics. *Cognition*, *106*(3), 1558–1568.

http://doi.org/10.1016/j.cognition.2007.06.010
Starkey, P., Spelke, E. S., & Gelman, R. (1990). Numerical abstraction by human infants. *Cognition*, *36*(2), 97–127. http://doi.org/10.1016/0010-0277(90)90001-Z
Téglás, E., Girotto, V., Gonzalez, M., & Bonatti, L. L. (2007). Intuitions of probabilities shape expectations about the future at 12 months and beyond. *Proceedings of the National Academy of Sciences of the United States of America*, *104*(48), 19156–9. http://doi.org/10.1073/pnas.0700271104
Teglas, E., Ibanez-Lillo, A., Costa, A., & Bonatti, L. L. (2015). Numerical representations and intuitions of probabilities at 12 months. *Developmental Science*, *2*, 183–193. http://doi.org/10.1111/desc.12196
Téglás, E., Vul, E., Girotto, V., Gonzalez, M., Tenenbaum, J. B., & Bonatti, L. L. (2011a). Pure reasoning in 12-month-old infants as probabilistic inference. *Science (New York, N.Y.)*, *332*(2011), 1054–1059. http://doi.org/10.1126/science.1196404
Téglás, E., Vul, E., Girotto, V., Gonzalez, M., Tenenbaum, J. B., & Bonatti, L. L. (2011b). Pure reasoning in 12-month-old infants as probabilistic inference. *Science (New York, N.Y.)*, *332*(6033), 1054–9. http://doi.org/10.1126/science.1196404
Tsuji, S., Bergmann, C., & Cristia, A. (2014). Community-augmented meta-analyses: Toward cumulative data assessment. *Perspectives on Psychological Science*, *9*, 661–665. http://doi.org/10.1177/1745691614552498
Turati, C., Bulf, H., & Simion, F. (2008). Newborns' face recognition over changes in viewpoint. *Cognition*, *106*(3), 1300–21. http://doi.org/10.1016/j.cognition.2007.06.005
U. S. Census Bureau. (2014a). Educational Attainment of the Population 18 Years and Over, by Age, Sex, Race, and Hispanic Origin: 2014. Retrieved January 12, 2016, from http://www.census.gov/hhes/socdemo/education/data/cps/2014/Table 1-01.xlsx
U. S. Census Bureau. (2014b). State & County QuickFacts. Retrieved February 20, 2016, from http://www.census.gov/quickfacts/table/PST045215/00
Valsecchi, M., Toscani, M., & Gegenfurtner, K. R. (2013). Perceived numerosity is reduced in peripheral vision. *Journal of Vision*, *13*(13), 7. http://doi.org/10.1167/13.13.7
Wood, J. N., & Spelke, E. S. (2005). Chronometric studies of numerical cognition in five-month-old infants. *Cognition*, *97*(1), 23–39. http://doi.org/10.1016/j.cognition.2004.06.007
Xu, F., & Garcia, V. (2008). Intuitive statistics by 8-month-old infants. *Proceedings of the National Academy of Sciences of the United States of America*, *105*(13), 5012–5. http://doi.org/10.1073/pnas.0704450105
Xu, F., Spelke, E. S., & Goddard, S. (2005). Number sense in human infants. *Developmental Science*, *8*(1), 88–101. http://doi.org/10.1111/j.1467-7687.2005.00395.x
Yakovlev, P. I., & Lecours, A.-R. (1967). The myelogenetic cycles of regional maturation of the brain. In *Regional Development of Brain in Early Life* (pp. 3–70).
Yu, C., & Smith, L. B. (2012). Embodied attention and word learning by toddlers. *Cognition*, *125*(2), 244–262. http://doi.org/10.1016/j.cognition.2012.06.016
Yuan, S., & Fisher, C. (2009). "Really ? She Blicked the Baby ?" Two-Years-Olds Learn Combinational Facts About Verbs by Listening. *Psychological Science*, *20*(5), 619–626. http://doi.org/10.1111/j.1467-9280.2009.02341.x

# Appendix: online repositories for data and code

**Chapter 2:** Raw data (video coding files and looking times per participant), stimuli and presentation code, and analysis code will be available upon publication in the Open Science Framework (OSF) repository https://osf.io/5fds4/. General code for processing VCode output using Matlab is available at https://github.com/kimberscott/matvcode.

**Chapters 3 and 4:** Raw data and analysis code are available in the OSF repository https://osf.io/9b4qj/. Code for the prototype Lookit site, including procedures and stimuli for test studies, is available at https://github.com/kimberscott/lookitprototype.

**Chapter 5:** The current lookit-api, ember-lookit-frameplayer, experimenter, and exp-addons repositories are available at https://github.com/CenterForOpenScience/. Documentation for writing studies using experimenter is available at http://lookit.readthedocs.io/en/latest/; for using the lookit API and experimenter site itself, at http://lookit-api.readthedocs.io/en/latest/; and specific frames are documented using yuidoc at http://centerforopenscience.github.io/exp-addons/modules/frames.html.