Injustice Attention Report

Tom Beesley

Most recent analysis run on: Tue Apr 29 15:07:37 2025

Setup and loading in the data

The data I'm loading in here is the fixation data for all participants. This is after having processed the raw data into fixations, which is handled in another script (stitch_data.R)

```r
library(tidyverse)
library(eyetools) # uses 0.9.1
library(effectsize)
library(afex)

citation("eyetools")
```

```
To cite eyetools in publications use:

  Tom Beesley and Matthew Ivory, (). {eyetools}: Tools for eye data
  analysis. R package version 0.9.1.
  https://tombeesley.github.io/eyetools/

A BibTeX entry for LaTeX users is

  @Manual{,
    title = {{eyetools}: Tools for eye data analysis},
    author = {Tom Beesley and Matthew Ivory},
    note = {R package version 0.9.1},
    url = {https://tombeesley.github.io/eyetools/},
  }
```

```r
rm(list = ls())

load("inj_unsw_EG_data.RData")

# limit to period 2 - stimulus presentation period

fixation_data <- filter(fixation_data, period == 2)

# limit fixations to within bounds of the screen

fixation_data <-
  fixation_data %>%
  filter(between(x, 0, 1920) & between(y,0,1080))
```

Setting up AOIs

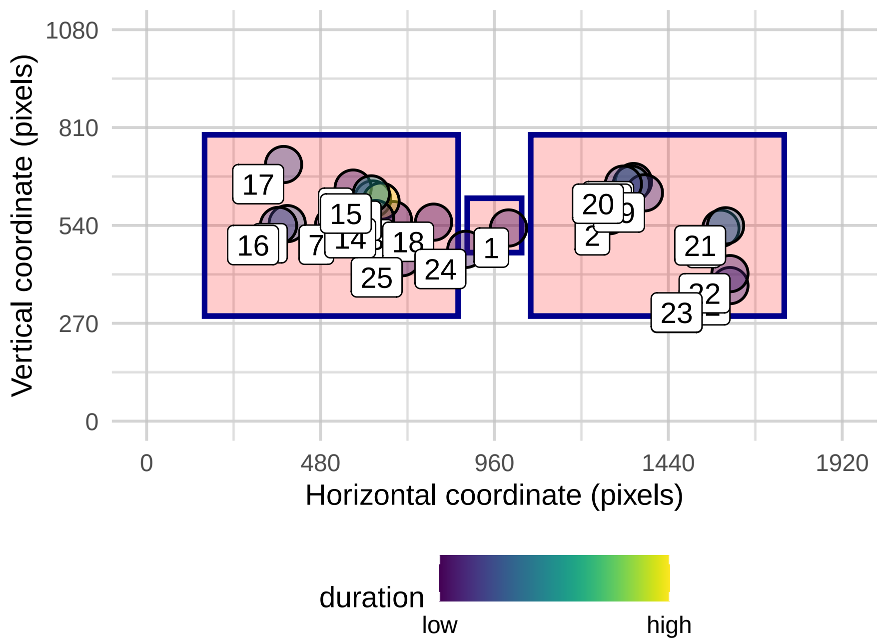
I set up three AOIs, for left, fixation, and right. Left and right AOIs are now calculated from the original matlab code.

```r
# Set AOIs
AOI_stims <- eyetools::create_AOI_df(3)

AOI_stims[1,] <- c(510, 540, 700, 500) # X, Y, W, H - left
AOI_stims[2,] <- c(1410, 540, 700, 500) # X, Y, W, H - right
AOI_stims[3,] <- c(960, 540, 150, 150) # X, Y, W, H - fixation

AOI_stims_names <- c("left", "right", "centre")
```

Number of fixations

Here's what a typical trial looks like with 25 fixations (using `eyetools::spatial_plot()`):

```r
eyetools::plot_spatial(fix_data = fixation_data,
                       pID_values = "001",
                       trial_values = 3,
                       AOIs = AOI_stims)
```



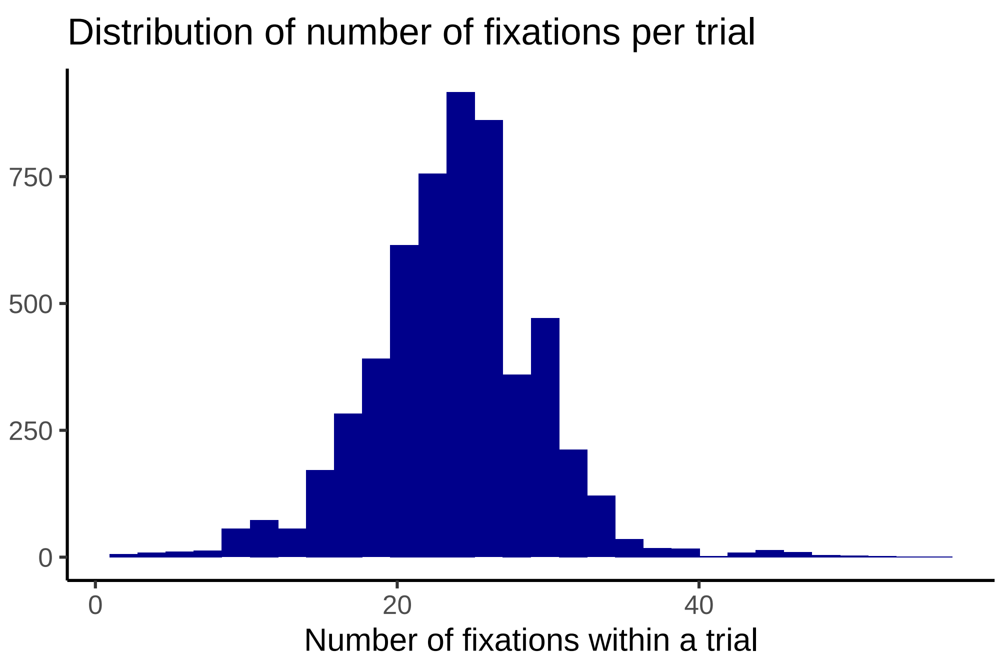
You can see from the index numbers that there is back and forth processing between the two stimuli. We can use `eyetools::AOI_seq()` to look at how the eye movements entered the different AOIs (here, just for the first 5 trials for this participant):

```
data_p1 <-
  fixation_data %>%
  filter(pID == "001")

eyetools::AOI_seq(data = data_p1,
                  AOIs = AOI_stims,
                  AOI_names = AOI_stims_names) %>%
  head(5)
```

```
  pID trial    AOI start  end duration entry_n
1 001     1  right   437 1337      900       1
2 001     1  right  1883 2680      797       2
3 001     1   left  2726 5346     2620       3
4 001     1  right  5393 8545     3152       4
5 001     1 centre  8555 8925      370       5
```

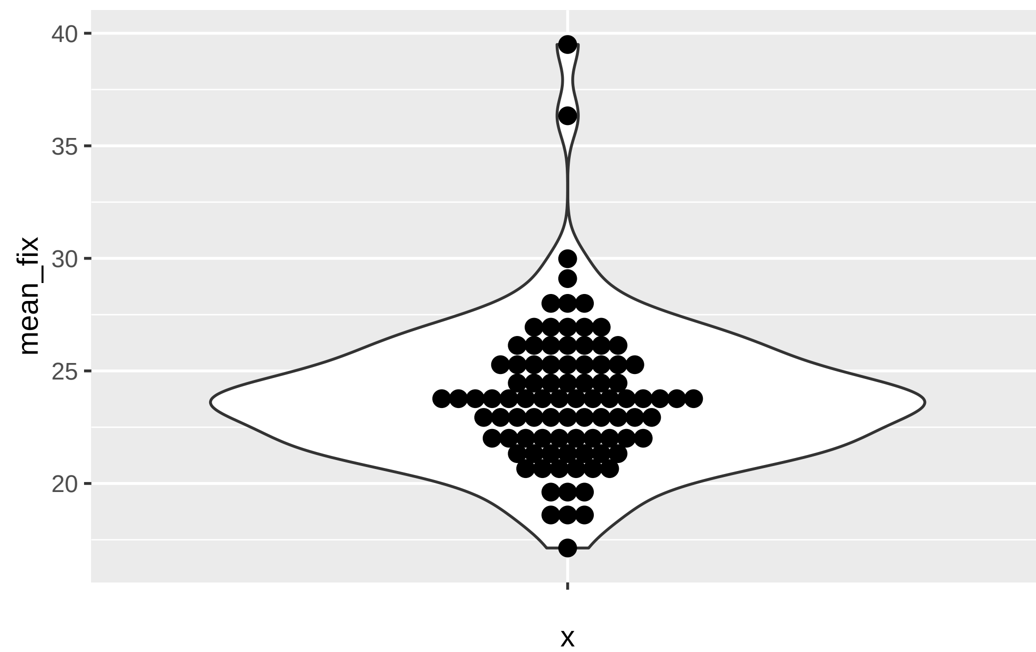
Here's the distribution of the number of fixations on a trial for all the data (for all participants):

```
fixation_data %>%
  group_by(pID, trial) %>%
  summarise(nFix = n()) %>%
  ggplot() +
  geom_histogram(aes(nFix), bins = 30, fill = "dark blue") +
  theme_classic(base_size = 12) +
  labs(title = "Distribution of number of fixations per trial",
       x = "Number of fixations within a trial",
       y = "")
```



Do the participants vary much in the mean number of fixations per trial? There are a couple of outliers in terms of number of fixations.

```
fixation_data %>%
  group_by(pID, trial) %>%
  summarise(n_fix = n()) %>%
  group_by(pID) %>%
  summarise(mean_fix = mean(n_fix)) %>%
  ggplot(aes(mean_fix, x = "")) +
  geom_violin() +
  geom_dotplot(binaxis = "y", stackdir = "center")
```



Looking at time in AOIs for the different trial types / stimulus valence

Following that exploration, we can take a look at the more important stuff:

The following code takes the pair types that you've defined and matches them up with the trial data for each participant. Thus we end up with a dataframe that contains the time spent in each area of interest, and what was presented on the screen. Then depending on the location of the target (which I take to be the first stimulus named in each pair) we take the data from AOI_1 and AOI_3. This means that we then have columns that reflect time on "Stim 1" (the first stimulus in each pair) and "Stim 2" (the second stimulus in each pair), irrespective of whether they are on the left or right each trial.

```
stim_codes <-
  read_csv("stim_details.csv") %>%
  rename(trial_ID = pair_num)


id_data <-
  id_data %>%
  left_join(stim_codes, by = "trial_ID") %>%
  rename(pID = pNum) %>%
  group_by(pID) %>%
  mutate(trial = 1:60) %>%
```

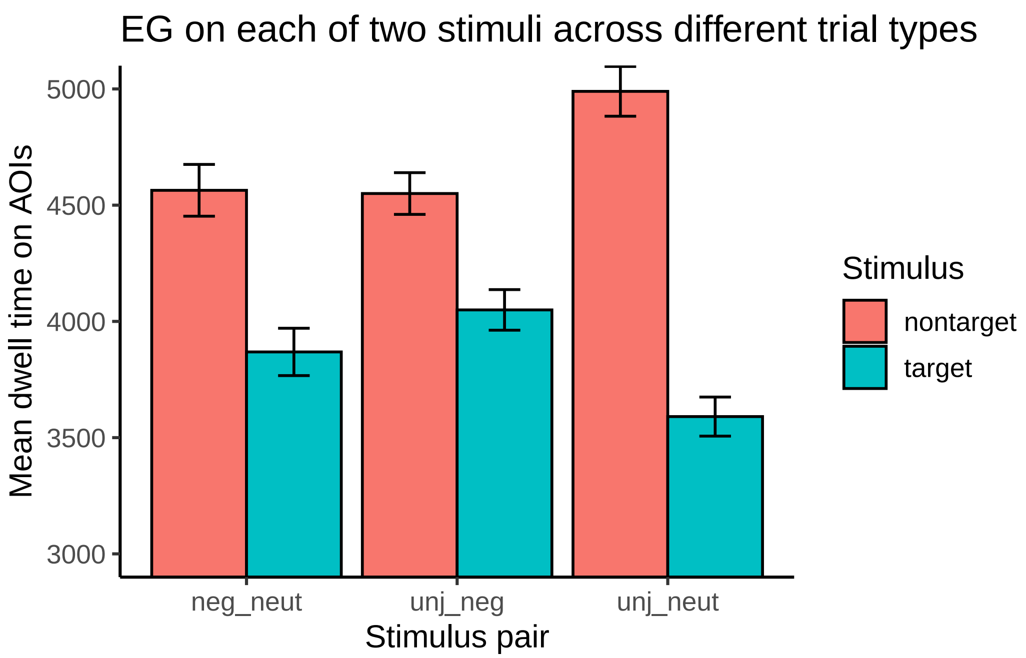
```
  ungroup()

# perform a conditional transform of the fixation data across the horiztonal midline
fixation_data <-
  left_join(fixation_data, id_data, by = c("pID", "trial"))

fixation_data <-
  conditional_transform(data = fixation_data,
                        flip = "x",
                        cond_column = "target_loc",
                        cond_values = 1)


AOI_time_df <- AOI_time(fixation_data, data_type = "fix", AOIs = AOI_stims)

eg_data_combined <-
  left_join(id_data, AOI_time_df, by = c("pID", "trial")) %>%  # bind together the data
  drop_na(AOI) %>%
  pivot_wider(names_from = AOI, values_from = time)

# pair type
# 1 = UNJ-NEG
# 2 = UNJ-NEUT
# 3 = NEG-NEUT

# target_loc
# 1 = LHS
# 2 = RHS

eg_data_full <-
  eg_data_combined %>%
  mutate(pair_type = case_match(pair_type,
                                1 ~ "unj_neg",
                                2 ~ "unj_neut",
                                3 ~ "neg_neut")) %>%
  rename(target_time = AOI_1,
         nontarget_time = AOI_2,
         centre_time = AOI_3)
```

Summarising the data for visualisation

Taking this final combined dataframe, we can summarise how much time each person spends on each stimulus in the pair, as a function of the trial type. This is plotted below:

```
# summarise the data for each participant, to get
# the critical 6 values (3 pairs x 2 stimuli)
eg_means_summary <-
  eg_data_full %>%
  group_by(pID, pair_type) %>%
  summarise(mean_target = mean(target_time, na.rm = TRUE),
            mean_nontarget = mean(nontarget_time, na.rm = TRUE))

# rather than have two columns for the two stimuli
# we should pivot the data and make stimulus a variable
eg_means_summary <-
eg_means_summary %>%
  pivot_longer(cols = c("mean_target", "mean_nontarget"),
               names_prefix = "mean_",
               names_to = "stimulus",
               values_to = "EG_time")

# calculate grand means and SE for the sample (for plotting)
eg_fig <-
  eg_means_summary %>%
  group_by(pair_type, stimulus) %>%
  summarise(meanEG = mean(EG_time),
            SE_EG = sd(EG_time)/sqrt(n()))

# plot the data
eg_fig %>%
  ggplot(aes(x = pair_type, y = meanEG,
             fill = stimulus,
             ymin = meanEG - SE_EG,
             ymax = meanEG + SE_EG)) +
  geom_col(color= "black",
           position=position_dodge()) +
  geom_errorbar(position = position_dodge(.9),
                width = .3) +
  coord_cartesian(ylim = c(3000, 5000)) +
  theme_classic(base_size = 12) +
  labs(y = "Mean dwell time on AOIs",
```

```
      x = "Stimulus pair",
      fill = "Stimulus",
      title = "EG on each of two stimuli across different trial types")
```



We'll run a t-test to see if the difference between unjust and negative stimuli is a real effect - yes it is! And with a medium effect size.

```
# filter to just the UNJ-NEG data
t_data_unj_neg <-
  eg_means_summary %>%
  filter(pair_type == "unj_neg")

# paired t-test
t.test(t_data_unj_neg$EG_time[t_data_unj_neg$stimulus == "target"],
       t_data_unj_neg$EG_time[t_data_unj_neg$stimulus == "nontarget"],
       paired = TRUE)
```

```
    Paired t-test

data:  t_data_unj_neg$EG_time[t_data_unj_neg$stimulus == "target"] and t_data_unj_neg$EG_time
t = -5.2321, df = 91, p-value = 1.066e-06
```

```
alternative hypothesis: true mean difference is not equal to 0
95 percent confidence interval:
 -690.6753 -310.5580
sample estimates:
mean difference
      -500.6166
```

```
# relevant effect size
effectsize::cohens_d(t_data_unj_neg$EG_time[t_data_unj_neg$stimulus == "target"],
                     t_data_unj_neg$EG_time[t_data_unj_neg$stimulus == "nontarget"],
                     paired = TRUE)
```

```
Cohen's d |         95% CI
--------------------------
-0.55     | [-0.76, -0.33]
```

Proportion of trials with a first fixation on target/non-target

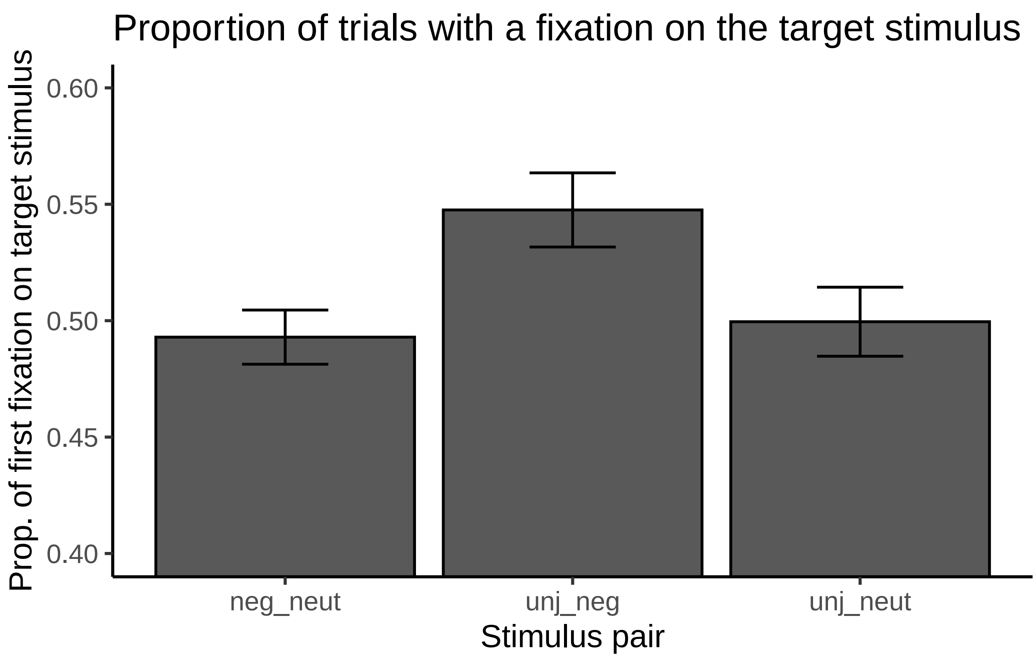
```
# get AOI seq result for all participants
AOI_seq_df <- AOI_seq(fixation_data, AOIs = AOI_stims, AOI_names = AOI_stims_names)
eg_seq_data_combined <- left_join(id_data, AOI_seq_df, by = c("pID", "trial")) # bind togethe

# for each trial get details of the first fixation on each AOI
first_entry_each_AOI <-
  eg_seq_data_combined %>%
  group_by(pID, trial) %>%
  distinct(AOI, .keep_all = TRUE) %>%
  mutate(pair_type = case_match(pair_type,
                                1 ~ "unj_neg",
                                2 ~ "unj_neut",
                                3 ~ "neg_neut")) %>%
  mutate(AOI = case_match(AOI,
                          "left" ~ "target",
                          "right" ~ "nontarget",
                          "centre" ~ "centre")) %>%
  ungroup() %>%
  complete(trial, AOI) %>%
  arrange(pID, trial, entry_n)
```

```r
write_csv(first_entry_each_AOI, file = "first_entry_time.csv")

# pair type
# 1 = UNJ-NEG
# 2 = UNJ-NEUT
# 3 = NEG-NEUT

# target_loc
# 1 = LHS
# 2 = RHS

# create variable reflecting centre fixation as first entry
first_entry_each_AOI <-
  first_entry_each_AOI %>%
  group_by(pID, trial) %>%
  mutate(centreFirst = case_when(AOI[1L] == "centre" ~ 1,
                                 .default = 0),
         firstStim = case_when(AOI[1L] == "centre" & n() > 1 ~ AOI[2L],
                               .default = NA))

# proportion of trials overall with a fixation on the centre
mean(first_entry_each_AOI$centreFirst, na.rm = TRUE)
```

```
[1] 0.7887649
```

```r
pair_summary_first_EG <-
  first_entry_each_AOI %>%
  filter(centreFirst == 1) %>%
  group_by(pID, trial) %>%
  slice(1) %>%
  group_by(pID, pair_type) %>%
  summarise(prop_target_first = mean(firstStim=="target", na.rm = TRUE))


pair_summary_first_EG %>%
  group_by(pair_type) %>%
  summarise(prop_targ = mean(prop_target_first),
            se = sd(prop_target_first)/sqrt(n())) %>%
  ggplot(aes(x = pair_type, y = prop_targ,
             ymin = prop_targ - se,
             ymax = prop_targ + se)) +
```

```r
  geom_col(color= "black",
           position=position_dodge()) +
  geom_errorbar(position = position_dodge(.9),
                width = .3) +
  coord_cartesian(ylim = c(0.4, 0.6)) +
  theme_classic(base_size = 12) +
  labs(y = "Prop. of first fixation on target stimulus",
       x = "Stimulus pair",
       title = "Proportion of trials with a fixation on the target stimulus")
```




```r
aov_car(data = pair_summary_first_EG,
        formula = prop_target_first ~ Error(pID/pair_type))
```

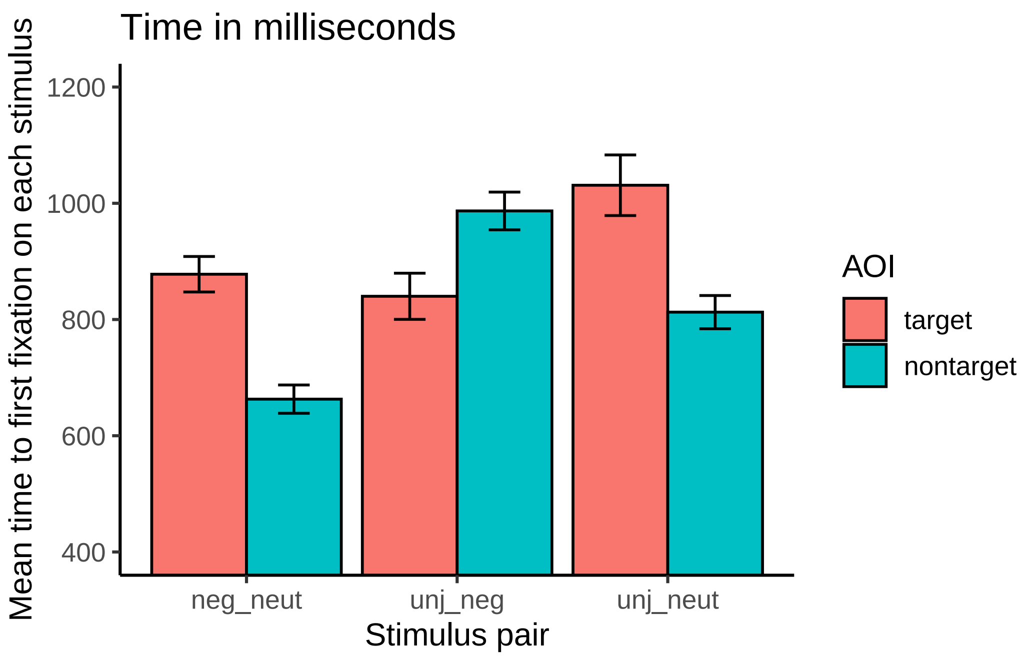
```
Anova Table (Type 3 tests)

Response: prop_target_first
     Effect           df  MSE       F  ges p.value
1 pair_type 1.85, 166.73 0.02 6.23 ** .041    .003
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1

Sphericity correction method: GG
```

In this analysis I looked at the order in which participants landed in the three different areas of interest - left stimulus, right stimulus, and central fixation. I then took just the initial entries on each trial, noting when participants looked at the centre first (i.e., were following instructions to fixate on the fixation cross). On average, participants did this on 79% of trials, and I analysed only data from these trials (the pattern is the same if you include trials where they didn't fixate first). Then I noted which stimulus they looked at first. From this we can calculate the proportion of trials on which they fixated the target or the alternative stimulus. It is fairly close to 50/50 in all three trial types, but there is some suggestion of a difference on the UNJ v NEG trial type. Here they seem to go to the target stimulus on about 55% of trials. The ANOVA reveals an effect of trial type.

Time to first fixation on each stimulus

```
time_of_first_entry <-
  first_entry_each_AOI %>%
  filter(centreFirst == 1) %>%
  group_by(pID, trial, pair_type, AOI) %>%
  summarise(timeOn = start)

timeOn_by_P <-
  time_of_first_entry %>%
  filter(AOI != "centre") %>%
  mutate(AOI = fct_relevel(AOI, "target", "nontarget")) %>%
  group_by(pID, pair_type, AOI) %>%
  summarise(timeOn = mean(timeOn, na.rm = TRUE)) %>%
  ungroup() %>%
  complete(pID, pair_type, AOI) # complete missing values

write_csv(timeOn_by_P, file = "time_to_first_fixation_on_AOI.csv")

timeOn_by_P %>%
  group_by(pair_type, AOI) %>%
  summarise(mean_timeOn = mean(timeOn, na.rm = TRUE),
            se_timeOn = sd(timeOn, na.rm = TRUE)/sqrt(n())) %>%
  ggplot(aes(x = pair_type, fill = AOI, y = mean_timeOn,
             ymin = mean_timeOn - se_timeOn,
             ymax = mean_timeOn + se_timeOn)) +
  geom_col(position=position_dodge(), colour = "black") +
  geom_errorbar(position = position_dodge(.9),
                width = .3) +
```

```r
  coord_cartesian(ylim = c(400, 1200)) +
  theme_classic(base_size = 12) +
  labs(y = "Mean time to first fixation on each stimulus",
       x = "Stimulus pair",
       title = "Time in milliseconds")
```




```r
aov_car(data = timeOn_by_P,
        formula = timeOn ~ Error(pID/pair_type*AOI))
```

```
Anova Table (Type 3 tests)

Response: timeOn
        Effect            df       MSE         F  ges p.value
1     pair_type 1.90, 171.16  63733.89 22.12 *** .041   <.001
2           AOI        1, 90 125621.84   9.24 ** .018    .003
3 pair_type:AOI 1.93, 173.80 114965.49 19.03 *** .062   <.001
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1

Sphericity correction method: GG
```

This figure shows the average time it takes to fixate on the target (first stimulus in the pair) versus the non-target (second stimulus in the pair) as a function of pair type.