

بسم تعالی



آزمایشگاه علوم اعصاب

تمرین شماره ۲

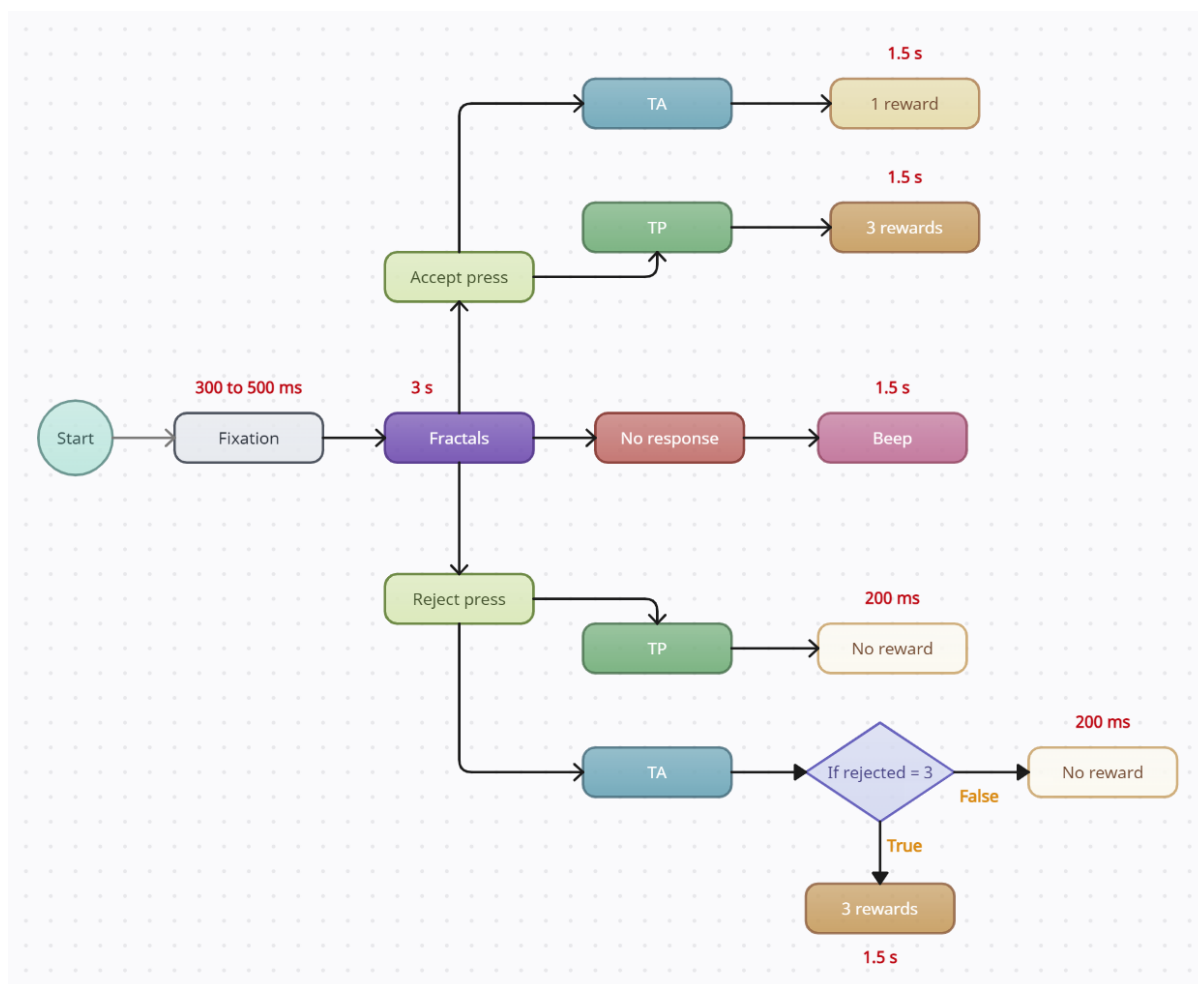
امیرحسین زاهدی ۹۹۱۰۱۷۰۵

بهار ۱۴۰۳

برنامه طراحی شده بر اساس آزمایشی تولید شده است که در آن جست و جوی تصویر بر اساس ارزش مورد بررسی قرار می گیرد به صورتی که بررسی می شود آیا اگر تصویر ارزشمندی در میان تصاویر مختلف به نمایش در آید، باعث سوق یافتن تمرکز و جست و جوی چشم به سمت آن می شود یا خیر. البته که این آزمایش می تواند نتایج متفاوتی را به دنبال داشته باشد. در این تکلیف سعی شده است تا برنامه ای بر پایه متلب و با استفاده از psychtoolbox طراحی شود تا در آن افراد مورد آزمایش بتوانند عملکرد خود را ثبت کنند.

بخش اول:

در بخش اول ابتدا قبل از رجوع به متلب، سعی می شود تا دیاگرام حالت آزمایش طراحی شود تا روند انجام آزمایش منظم تر و بهتر انجام شود. دیاگرام حالت به صورت زیر است:



در ابتدای آزمایش نقطه ای بر روی صفحه به مدت ۳۰۰ الی ۵۰۰ میلی ثانیه به صورت رندوم نمایش داده می شود. سپس فرکتال های آن تریال به مدت ۳ ثانیه نمایش داده می شود. در هنگام نمایش فرکتال ها فرد مورد آزمایش انتخاب می کند که آیا فرکتال ارزشمندی در میان فرکتال ها وجود دارد که انتخاب کند یا خیر. اگر انتخابی داشته باشد

وارد بخش بالا، و اگر تریال را رد کند وارد بخش پایین می شود. اگر اقدام مفیدی نکند، به مدت ۱.۵ ثانیه بوقی به صدا در می آید.

در بخش بالا اگر فرکتال ارزشمند وجود داشته باشد و درست انتخاب شود، ۳ جایزه داده می شود اما اگر وجود نداشته باشد یا اشتباه انتخاب شود، ۱ جایزه داده می شود. اهدای جایزه ۱.۵ ثانیه طول می کشد.

در بخش پایین در صورتی که ۳ بار به درستی رد نشده باشد، جایزه ای داده نمی شود و ۲۰۰ میلی ثانیه طول می کشد تا تریال بعدی نمایش داده شود. اما در صورتی که ۳ بار تریال به درستی رد شود، ۳ جایزه به فرد داده می شود.

بخش دوم:

۴۸ فرکتال موجود است که این ۴۸ فرکتال به صورت رندوم به دو گروه ۲۴ تایی value و perceptual تقسیم می شوند. هر کدام از این دو گروه ۲۴ تایی نیز به دو گروه ۱۲ تایی با ارزش و بی ارزش تقسیم می شوند. انتخاب رندوم فرکتال ها برای هر فرد به دلیل رفع بایاس است. دو حالت مینیم و ماکسیم دارد. به صورتی که در حالت مینیم می توانیم این ۴ گروه فرکتال را برای ۴ نفر جایگشت دهیم. به عبارتی مینیم تعداد افرادی که بتوانیم فرکتال ها در هر ۴ گروه باشند ۴ نفر است. اما اگر بخواهیم ماکسیم تعداد افرادی که بتوانند به صورت رندوم و بی تکرار فرکتال ها را تجربه کنند، بسنجیم، هر فرکتال می تواند در ۴ گروه باشد. پس به تعداد ۴ به توان ۴۸ حالت بی تکرار می توانیم فرکتال ها را بچینیم. کد این بخش به صورت زیر است:

```
%%%%%%%%%%%% Data grouping %%%%%%%%%%%%%%
% randomizing fractals
sorted_fractals = cell(48,1);
random_perm = randperm(length(all_fractals));
sorted_fractals = all_fractals(random_perm);
sorted_file_list = file_list(random_perm,:);

% making cells
value_good_img = cell(12,1);
perceptual_good_img = cell(12,1);
value_bad_img = cell(12,1);
perceptual_bad_img = cell(12,1);

% grouping images
value_good_img = sorted_fractals(1:12);
perceptual_good_img = sorted_fractals(13:24);
value_bad_img = sorted_fractals(25:36);
perceptual_bad_img = sorted_fractals(37:48);

% grouping lists
value_good_list = sorted_file_list(1:12,:);
perceptual_good_list = sorted_file_list(13:24,:);
value_bad_list = sorted_file_list(25:36,:);
perceptual_bad_list = sorted_file_list(37:48,:);
```

بخش سوم:

پس از بخش بندی به ۴ گروه، تریال ها را به صورت رندوم تنظیم می کنیم به صورت که تعداد تریال ها به ۴ گروه مساوی تقسیم می شوند و فرکتال های خوب نیز به تعداد مساوی در تریال ها پخش می شوند. تریال های نا خوب نیز به صورت غیر تکراری در تریال ها پخش می شوند. کد این بخش نیز به صورت زیر است:

```
%%%%%%%%%%%%%% Trials making %%%%%%%%%%%%%%%
% randomizing trials display size
trials_ds =
[3*ones(1, trials_number/4), 5*ones(1, trials_number/4), 7*ones(1, trials_number/4), 9*ones(1, trials_number/4)].';
trials_ds = trials_ds(randperm(trials_number));

% randomizing trials present or absent
trials_ap =
[ones(1, trials_number/4), 2*ones(1, trials_number/4), 3*ones(1, trials_number/4), 4*ones(1, trials_number/4)].';
trials_ap = trials_ap(randperm(trials_number));

% randomizing trials degree
trials_degree = randi([1, 360], trials_number, 1);

% randomizing good fractals
a = 1:12;
a = [a,a,a].';
value_good_fractals_list = a(randperm(36));
perceptual_good_fractals_list = a(randperm(36));
```

بخش چهارم:

در ابتدا اسم، آیدی، شماره بلاک، تعداد تریال در بلاک و میزان فاصله چشم از صفحه از آزمودنی دریافت می شود که کد آن به صورت زیر است:

```
%%%%%%%%%%%%%% Subject inputs %%%%%%%%%%%%%%%
subject_name = input("Please enter your name:", "s");
subject_ID = input("Please enter your ID:");
session_number = input("Please enter session number:");
trials_number = input("How many trials are you considering?:");
distance = input("How distant is you eye from monitor in cm?:");
```

مراحل انجام آزمایش همانند فیلم داده شده طراحی می شوند به صورتی که حرکت موس به مانند حرکت چشم در هر لحظه دریافت می شود و با حذف مکان های تکراری، مسیر موس به صورت بهینه بدست می آید.

در هر تریال با توجه به تعداد فرکتال های آن تریال که به صورت رندوم و مساوی چیده شده اند، با در نظر گرفتن روتیشن رندوم ۱ تا ۳۶۰ درجه، فرکتال ها به صورت منظم چیده می شوند. دور فرکتال ارزشمند مربع سبز قرار می گیرد ولی به دور فرکتال های کم ارزش مربعی قرار نمی گیرد.

در هر تریال فرد می تواند با فشردن دکمه R تریال را ریجکت کند و یا با موس بر روی فرکتال مورد نظر کلیک کند. کلیک کردن بر روی فضای خالی به معنای عدم پاسخ است و عدم پاسخ پس از ۳ ثانیه باعث بوق زدن به مدت ۱.۵ ثانیه می شود. همچنین فرد می تواند با فشردن Q در هر جای آزمایش به آن پایان دهد. کد این بخش که عمده برنامه است در زیر مشهود است. بخش های مربوط به جایزه توضیح داده خواهند شد.

```
for i = 1:trials_number

    %%%%%%%%%%% Center scene %%%%%%%%%%%
    duration_fix = randi([300 500],1)/1000;
    waitframes_fix = round(duration_fix / ifi);
    time_fix = GetSecs;
    Screen('DrawDots', window, [xCenter, yCenter], 20, [255 255 255], [], 2);
    Screen('Flip', window);
    Screen('Flip', window, time_fix + (waitframes_fix) * ifi);

    %%%%%%%%%%% Image locations %%%%%%%%%%%
    R = 650;
    W = 350;
    loc = zeros(trials_ds(i),2);
    for j = 1:trials_ds(i)
        degree = (360/trials_ds(i))*j + trials_degree(i);
        degree = deg2rad(degree);
        loc(j,1) = round(R*cos(degree)) + xCenter;
        loc(j,2) = round(R*sin(degree)) + yCenter;
    end

    %%%%%%%%%%% Defining second part of outputs %%%%%%%%%%%
    output.fractalSize = atan((W*0.025/2)/distance)*180/pi;
    output.peripheralCircuit = atan((R*0.025)/distance)*180/pi;
    output.screenSize = windowRect(3:4);

    %%%%%%%%%%% Image defining & setting %%%%%%%%%%%
    images = cell(trials_ds(i),1);
    switch trials_ap(i)
        case 1
            t_value = t_value + 1;
            images{1} = value_good_img{value_good_fractals_list(t_value)};
            output.trials.fractalsName{i,1} =
value_good_list(value_good_fractals_list(t_value)).name;
            imageTexture = Screen('MakeTexture', window, images{1});
            Screen('DrawTexture', window, imageTexture, [], [loc(1,1)-W/2
loc(1,2)-W/2 loc(1,1)+W/2 loc(1,2)+W/2], 0);
            Screen('FrameRect', window, [0,255,0],[loc(1,1)-W/2 loc(1,2)-W/2
loc(1,1)+W/2 loc(1,2)+W/2],5);

            rp = randperm(12,trials_ds(i));
            for j = 2:trials_ds(i)
                images{j} = value_bad_img{rp(j)};
                output.trials.fractalsName{i,j} = perceptual_bad_list(rp(j)).name;
                imageTexture = Screen('MakeTexture', window, images{j});
                Screen('DrawTexture', window, imageTexture, [], [loc(j,1)-W/2
loc(j,2)-W/2 loc(j,1)+W/2 loc(j,2)+W/2], 0);
            end
        case 2
            rp = randperm(12,trials_ds(i));
            for j = 1:trials_ds(i)
```

```

        images{j} = value_bad_img{rp(j)};
        output.trials.fractalsName{i,j} = value_bad_list(rp(j)).name;
        imageTexture = Screen('MakeTexture', window, images{j});
        Screen('DrawTexture', window, imageTexture, [], [loc(j,1)-W/2
loc(j,2)-W/2 loc(j,1)+W/2 loc(j,2)+W/2], 0);
    end
    case 3
        t_perceptual = t_perceptual + 1;
        images{1} =
value_good_img{perceptual_good_fractals_list(t_perceptual),1};
        output.trials.fractalsName{i,1} =
perceptual_good_list(perceptual_good_fractals_list(t_perceptual)).name;
        imageTexture = Screen('MakeTexture', window, images{1});
        Screen('DrawTexture', window, imageTexture, [], [loc(1,1)-W/2
loc(1,2)-W/2 loc(1,1)+W/2 loc(1,2)+W/2], 0);
        Screen('FrameRect', window, [0,255,0],[loc(1,1)-W/2 loc(1,2)-W/2
loc(1,1)+W/2 loc(1,2)+W/2],5);

        rp = randperm(12, trials_ds(i));
        for j = 2:trials_ds(i)
            images{j} = perceptual_bad_img{rp(j)};
            output.trials.fractalsName{i,j} = perceptual_bad_list(rp(j)).name;
            imageTexture = Screen('MakeTexture', window, images{j});
            Screen('DrawTexture', window, imageTexture, [], [loc(j,1)-W/2
loc(j,2)-W/2 loc(j,1)+W/2 loc(j,2)+W/2], 0);
        end
        otherwise
            rp = randperm(12, trials_ds(i));
            for j = 1:trials_ds(i)
                images{j} = perceptual_bad_img{rp(j)};
                output.trials.fractalsName{i,j} = perceptual_bad_list(rp(j)).name;
                imageTexture = Screen('MakeTexture', window, images{j});
                Screen('DrawTexture', window, imageTexture, [], [loc(j,1)-W/2
loc(j,2)-W/2 loc(j,1)+W/2 loc(j,2)+W/2], 0);
            end
        end

        %%%%%%%%%%%%%%% Image showing %%%%%%%%%%%%%%%
        Screen('DrawDots', window, [xCenter, yCenter], 20, [255 255 255], [], 2);
        time_img = GetSecs;
        Screen('Flip', window, [], 1);

        %%%%%%%%%%%%%%% Response scene %%%%%%%%%%%%%%%
        error_flag = 0;
        clicked = 0;
        reward = 0;
        ShowCursor;
        SetMouse(xCenter, yCenter, window);
        mouse_dots = zeros(0,2);
        mouse_dots_numbers = 0;
        while ~(key == kQ || key == kR || error_flag == 1 || clicked == 1)
            %%%%%%%%%%%%%%% Mouse click %%%%%%%%%%%%%%%
            mouse_dots_numbers = mouse_dots_numbers + 1;
            [x,y,buttons] = GetMouse();
            mouse_dots(mouse_dots_numbers,1) = x;
            mouse_dots(mouse_dots_numbers,2) = y;
            if (buttons(1) == 1)
                for j = 1:trials_ds(i)

```

```

        if (x>loc(j,1)-W/2 && x<loc(j,1)+W/2 && y>loc(j,2)-W/2 &&
y<loc(j,2)+W/2)
            clicked = 1;
            if j == 1 && (trials_ap(i) == 1 || trials_ap(i) == 3)
                reward = 1;
            end
        end
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Keyboard %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if (error_flag == 0)
    [ keyIsDown, ~, keyCode ] = KbCheck(-1);
    if (keyIsDown)
        key = find(keyCode);
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Not responding %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
delta = GetSecs-time_img;
if (delta > 3)
    error_flag = 1;
    key = kSp;
end
end
unique_mouse_dots = unique(mouse_dots,'rows','stable');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Analyzing responses %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Beep %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if (error_flag == 1)
    t_beep = GetSecs;
    sound(audData,audFreq);
    Screen('Flip', window);
    Screen('Flip', window);
    Screen('Flip', window, t_beep + (waitframes_ITI_reward)* ifi);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Reject %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
elseif (key == kR)
    if trials_ap(i) == 2 || trials_ap(i) == 4
        rejected_numbers = rejected_numbers + 1;
    else
        rejected_numbers = 0;
    end
    key = kSp;
    t_ITI_reject = GetSecs;
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Reject without reward %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    if rejected_numbers < 3
        switch trials_ap(i)
            case 1
                imageTexture = Screen('MakeTexture', window, value_img);
                Screen('DrawTexture', window, imageTexture, [], [150 100 500
350], 0);
            case 3
                imageTexture = Screen('MakeTexture', window, perceptual_img);
                Screen('DrawTexture', window, imageTexture, [], [150 100 500
350], 0);
            case 2
                imageTexture = Screen('MakeTexture', window, value_img);

```

```

        Screen('DrawTexture', window, imageTexture, [], [150 100 500
350], 0);
        case 4
            imageTexture = Screen('MakeTexture', window, perceptual_img);
            Screen('DrawTexture', window, imageTexture, [], [150 100 500
350], 0);
        end
        imageTexture = Screen('MakeTexture', window, R_img);
        Screen('DrawTexture', window, imageTexture, [], [150 250 500 500], 0);
        HideCursor;
        Screen('Flip', window);
        Screen('Flip', window, t_ITI_reject + (waitframes_ITI_reject)* ifi);

        %%%%%%%%%%% Reject with reward %%%%%%%%%%%
        else
            w_reward = 100;
            x_reward = 230;
            y_reward = 450;
            rejected_numbers = 0;
            switch trials_ap(i)
                case 1
                    imageTexture = Screen('MakeTexture', window, value_img);
                    Screen('DrawTexture', window, imageTexture, [], [150 100 500
350], 0);
                case 3
                    imageTexture = Screen('MakeTexture', window, perceptual_img);
                    Screen('DrawTexture', window, imageTexture, [], [150 100 500
350], 0);
                case 2
                    imageTexture = Screen('MakeTexture', window, value_img);
                    Screen('DrawTexture', window, imageTexture, [], [150 100 500
350], 0);
                case 4
                    imageTexture = Screen('MakeTexture', window, perceptual_img);
                    Screen('DrawTexture', window, imageTexture, [], [150 100 500
350], 0);
            end
            imageTexture = Screen('MakeTexture', window, R_img);
            Screen('DrawTexture', window, imageTexture, [], [150 250 500 500], 0);
            imageTexture = Screen('MakeTexture', window, reward_img);
            Screen('DrawTexture', window, imageTexture, [], [x_reward y_reward
x_reward+w_reward y_reward+w_reward], 0);
            Screen('DrawTexture', window, imageTexture, [], [x_reward+100 y_reward
x_reward+100+w_reward y_reward+w_reward], 0);
            Screen('DrawTexture', window, imageTexture, [], [x_reward+50
y_reward+100 x_reward+50+w_reward y_reward+100+w_reward], 0);
            HideCursor;
            Screen('Flip', window);
            Screen('Flip', window, t_ITI_reject + (waitframes_ITI_reward)* ifi);
        end
        output.trials.buttonPressed{i} = 'Reject';

        %%%%%%%%%%% Quit %%%%%%%%%%%
        elseif (key == kQ)
            clear Screen;

        %%%%%%%%%%% Click with reward %%%%%%%%%%%
        elseif reward == 1
            t_click = GetSecs;

```



```

w_reward = 100;
x_reward = 230;
y_reward = 450;
switch trials_ap(i)
    case 1
        imageTexture = Screen('MakeTexture', window, value_img);
        Screen('DrawTexture', window, imageTexture, [], [150 100 500 350],
0);

    case 3
        imageTexture = Screen('MakeTexture', window, perceptual_img);
        Screen('DrawTexture', window, imageTexture, [], [150 100 500 350],
0);

    case 2
        imageTexture = Screen('MakeTexture', window, value_img);
        Screen('DrawTexture', window, imageTexture, [], [150 100 500 350],
0);

    case 4
        imageTexture = Screen('MakeTexture', window, perceptual_img);
        Screen('DrawTexture', window, imageTexture, [], [150 100 500 350],
0);

end
imageTexture = Screen('MakeTexture', window, mouse_img);
Screen('DrawTexture', window, imageTexture, [], [150 250 500 500], 0);
imageTexture = Screen('MakeTexture', window, reward_img);
Screen('DrawTexture', window, imageTexture, [], [x_reward y_reward
x_reward+w_reward y_reward+w_reward], 0);
Screen('DrawTexture', window, imageTexture, [], [x_reward+100 y_reward
x_reward+100+w_reward y_reward+w_reward], 0);
Screen('DrawTexture', window, imageTexture, [], [x_reward+50 y_reward+100
x_reward+50+w_reward y_reward+100+w_reward], 0);
for j = 1:size(unique_mouse_dots,1)
    Screen('DrawDots', window, [unique_mouse_dots(j,1),
unique_mouse_dots(j,2)], 20, [255-mod(j,256) 255 mod(j,256)], [], 2);
end
HideCursor;
Screen('Flip', window);
Screen('Flip', window, t_click + (waitframes_ITI_reward)* ifi);
output.trials.buttonPressed{i} = 'Accept';

%%%%%%%%%%%% Click without reward %%%%%%%%%%
else
    t_click = GetSecs;
    w_reward = 100;
    x_reward = 230;
    y_reward = 450;
    switch trials_ap(i)
        case 1
            imageTexture = Screen('MakeTexture', window, value_img);
            Screen('DrawTexture', window, imageTexture, [], [150 100 500 350],
0);

        case 3
            imageTexture = Screen('MakeTexture', window, perceptual_img);
            Screen('DrawTexture', window, imageTexture, [], [150 100 500 350],
0);

        case 2
            imageTexture = Screen('MakeTexture', window, value_img);
            Screen('DrawTexture', window, imageTexture, [], [150 100 500 350],
0);

        case 4

```

```

        imageTexture = Screen('MakeTexture', window, perceptual_img);
        Screen('DrawTexture', window, imageTexture, [], [150 100 500 350],
0);
    end
    imageTexture = Screen('MakeTexture', window, mouse_img);
    Screen('DrawTexture', window, imageTexture, [], [150 250 500 500], 0);
    imageTexture = Screen('MakeTexture', window, reward_img);
    Screen('DrawTexture', window, imageTexture, [], [x_reward+50 y_reward
x_reward+50+w_reward y_reward+w_reward], 0);
    for j = 1:size(unique_mouse_dots,1)
        Screen('DrawDots', window, [unique_mouse_dots(j,1),
unique_mouse_dots(j,2)], 20, [255-mod(j,256) 255 mod(j,256)], [], 2);
    end
    HideCursor;
    Screen('Flip', window);
    Screen('Flip', window, t_click + (waitframes_ITI_reward)* ifi);
    output.trials.buttonPressed{i} = 'Accept';
end

%%%%%%%%%% Defining third part of outputs %%%%%%%%%%
output.trials.fractalsPosition{i} = loc;
output.trials.DS{i} = trials_ds(i);
switch trials_ap(i)
    case 1
        output.trials.valueORperceptual{i} = 'value';
        output.trials.absentORpresent{i} = 'TP';
    case 2
        output.trials.valueORperceptual{i} = 'value';
        output.trials.absentORpresent{i} = 'TA';
    case 3
        output.trials.valueORperceptual{i} = 'perceptual';
        output.trials.absentORpresent{i} = 'TP';
    case 4
        output.trials.valueORperceptual{i} = 'perceptual';
        output.trials.absentORpresent{i} = 'TA';
end
output.trials.mousePosition{i} = unique_mouse_dots;

end

```

بخش پنجم:

در این بخش که کد آن همراه با بخش قبل در بالا آمده است، تعدادی عکس طراحی شده است زیرا که تکست در این لپ تاپ به درستی به نمایش در نمی آمد و برای تمام بخش های متنی عکس تهیه شده است. در پایان هر ترایال در صورت بروز پاسخ، گروه value یا perceptual فرکتال های نمایش داده شده، کلید یا موس فشرده شده به ندر گوشه سمت چپ تصویر به نمایش در می آیند. اگر جایزه ای داده شده باشد نیز در سمت چپ تصویر به نمایش در می آید که نحوه اختصاص دادن تعداد جوایز در بخش اول مطرح شده است. همچنین راه طی شده توسط موس نیز نمایش داده می شود.

کد نمونه به صورت زیر است:

```

switch trials_ap(i)
    case 1
        imageTexture = Screen('MakeTexture', window, value_img);

```

```

        Screen('DrawTexture', window, imageTexture, [], [150 100 500 350],
0);
        case 3
            imageTexture = Screen('MakeTexture', window, perceptual_img);
            Screen('DrawTexture', window, imageTexture, [], [150 100 500 350],
0);
        case 2
            imageTexture = Screen('MakeTexture', window, value_img);
            Screen('DrawTexture', window, imageTexture, [], [150 100 500 350],
0);
        case 4
            imageTexture = Screen('MakeTexture', window, perceptual_img);
            Screen('DrawTexture', window, imageTexture, [], [150 100 500 350],
0);
        end
        imageTexture = Screen('MakeTexture', window, mouse_img);
        Screen('DrawTexture', window, imageTexture, [], [150 250 500 500], 0);
        imageTexture = Screen('MakeTexture', window, reward_img);
        Screen('DrawTexture', window, imageTexture, [], [x_reward y_reward
x_reward+w_reward y_reward+w_reward], 0);
        Screen('DrawTexture', window, imageTexture, [], [x_reward+100 y_reward
x_reward+100+w_reward y_reward+w_reward], 0);
        Screen('DrawTexture', window, imageTexture, [], [x_reward+50 y_reward+100
x_reward+50+w_reward y_reward+100+w_reward], 0);
        for j = 1:size(unique_mouse_dots,1)
            Screen('DrawDots', window, [unique_mouse_dots(j,1),
unique_mouse_dots(j,2)], 20, [255-mod(j,256) 255 mod(j,256)], [], 2);
        end
        HideCursor;
        Screen('Flip', window);
        Screen('Flip', window, t_click + (waitframes_ITI_reward)* ifi);

```

بخش ششم:

در نهایت نیز اطلاعات در فایل ذخیره می شوند و به صورت خروجی ثبت می شوند. در این فایل اطلاعات کاربر، درجه بینایی برای هر فرکتال و نقطه مرکزی تصویر و ابعاد تصویر نیز ذخیره می شود. همچنین اطلاعات مربوط به هر تریال اعم از نوع فرکتال ها، رد یا قبول تریال، اسم فرکتال ها، مکان هر فرکتال در تصویر، تعداد فرکتال ها و مسیر موس ذخیره می شوند. بخش از کد در متن دیگر کد هاست اما بخشی که جداست به صورت زیر است. البته این سه بخش نیز در مکان های مختلف هستند.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Defining first part of outputs %%%%%%%%%%%%%%%
output = struct;
output.subjectName = subject_name;
output.subjectID = subject_ID;
output.sessionNumber = session_number;
output.trials = struct;
output.trials.buttonPressed = cell(trials_number,1);
output.trials.fractalsName = cell(trials_number,9);
output.trials.fractalsPosition = cell(trials_number,1);
output.trials.DS = cell(trials_number,1);
output.trials.valueORperceptual = cell(trials_number,1);
output.trials.absentORpresent = cell(trials_number,1);
output.trials.mousePosition = cell(trials_number,1);

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Defining second part of outputs %%%%%%%%%%%%%
output.fractalSize = atan((W*0.025/2)/distance)*180/pi;
output.peripheralCircuit = atan((R*0.025)/distance)*180/pi;
output.screenSize = windowRect(3:4);

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Defining third part of outputs %%%%%%%%%%%%%
output.trials.fractalsPosition{i} = loc;
output.trials.DS{i} = trials_ds(i);
switch trials_ap(i)
    case 1
        output.trials.valueORperceptual{i} = 'value';
        output.trials.absentORpresent{i} = 'TP';
    case 2
        output.trials.valueORperceptual{i} = 'value';
        output.trials.absentORpresent{i} = 'TA';
    case 3
        output.trials.valueORperceptual{i} = 'perceptual';
        output.trials.absentORpresent{i} = 'TP';
    case 4
        output.trials.valueORperceptual{i} = 'perceptual';
        output.trials.absentORpresent{i} = 'TA';
end
output.trials.mousePosition{i} = unique_mouse_dots;

```

بخش هفتم:

در حالتی که جایزه ای داده می شود یا پاسخی داده نمی شود، ۱.۵ ثانیه طول می کشد اما در حالت رد ترایال ۲۰۰ میلی ثانیه طول می کشد. کوتاهی زمان بین ترایال ها برای رد کردن، به این دلیل است که آزمودنی ترجیح می دهد تا به جای دریافت ۱ جایزه و صرف ۱.۵ ثانیه زمان، زود تر ترایال های بی ارزش را عبور دهد و به ترایال ارزشمند با ۳ جایزه برسد. تفاوت اختلاف ITI ها به این دلیل است. به نوعی این تفاوت می تواند استراتژی آزمودنی را نیز به صورت جالبی آشکار کند.

در تمام بخش های کد، کامنت گذاری و بخش بندی به نحو مناسبی انجام شده است که سبب سادگی در بررسی کد توسط فردی غیر از طراح است. البته در فیلم ضبط شده نیز مراحل کار توضیح داده شده است.