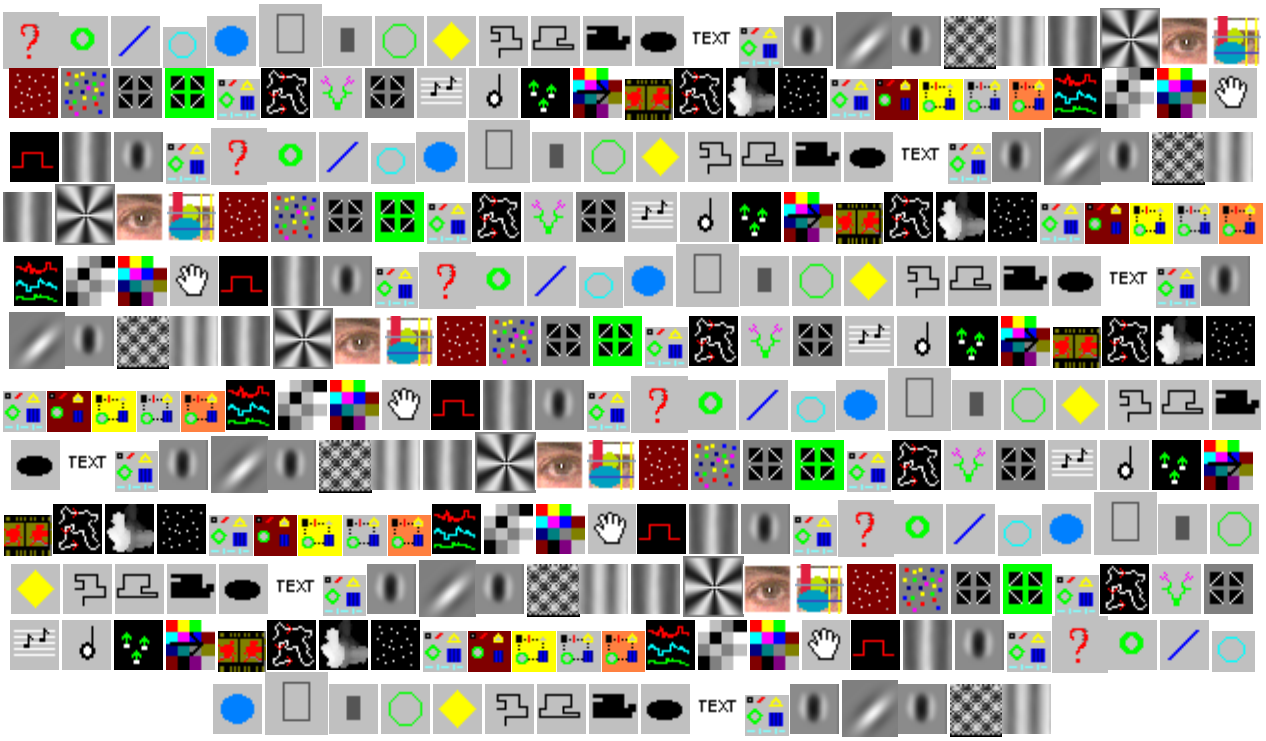
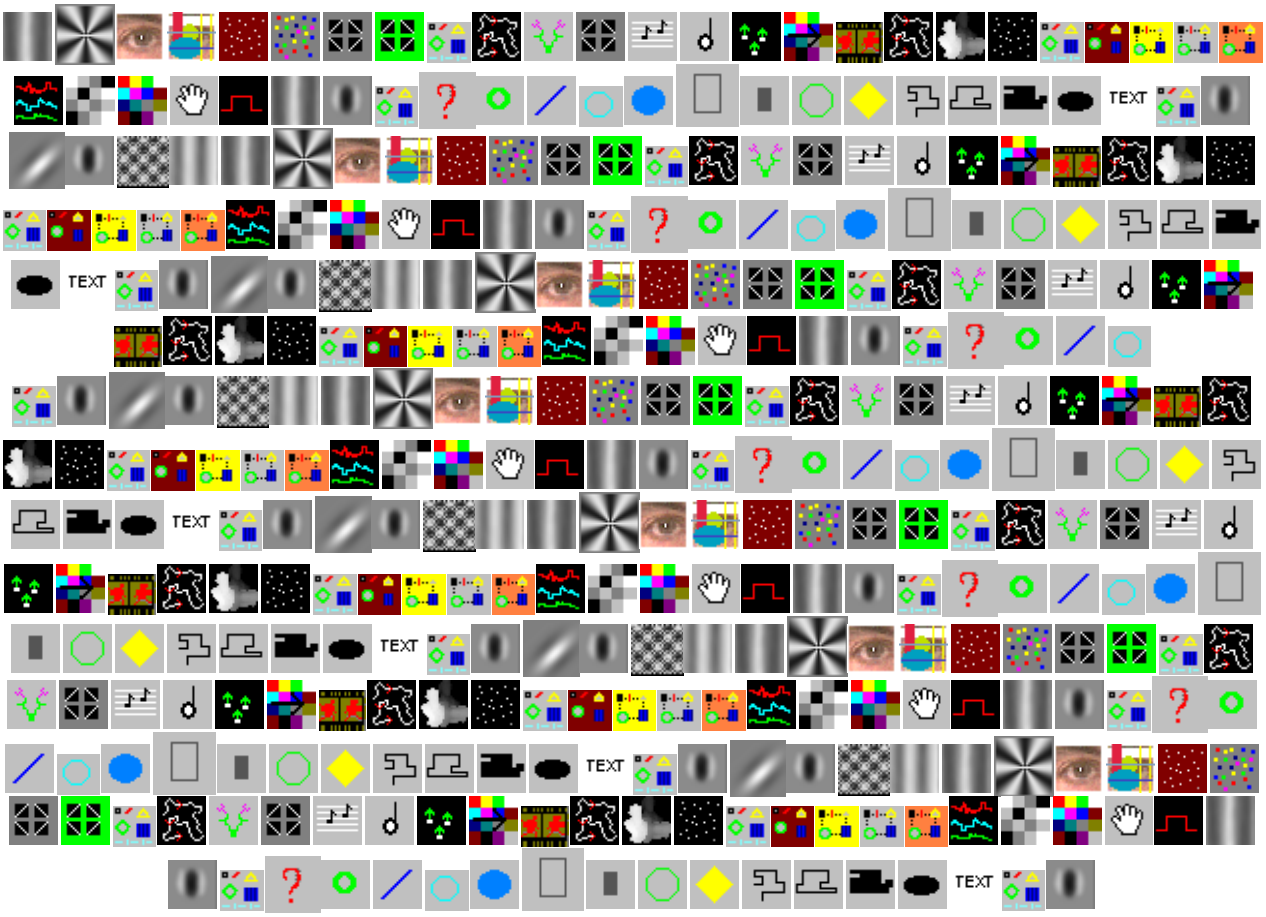


JEDA

User guide

2020



INTRODUCTION

Jeda is a software running under Windows (XP, W7, W10) using DirectX (directDraw version 7.0; DirectSound, DirectInput).
It is designed to edit a variety of stimuli and to use them in experiments to study visual cognition. It is written in C++ using CBuilder v1.0 (32 bit program, Borland).

Jeda software relies on managing a list of **Objects** (Graphics, Animation, Palette, Sounds, etc.), organized in a double chain list, as shown in Figure 1. Each Object has an ID code (its number) and, when needed, a "suitcase" of parameters adapted to each Object Type (Figure 2). The current Object "ACTU" can be edited at any time to change its characteristics. Many tools to move in the list are available.

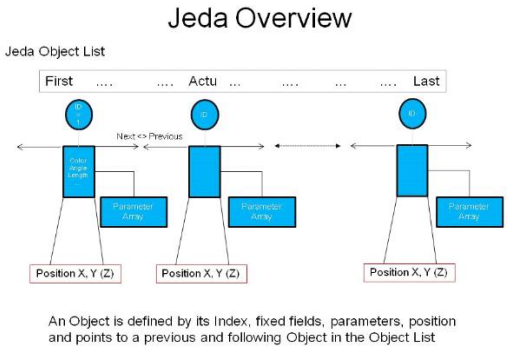


Figure 1: Double chain list

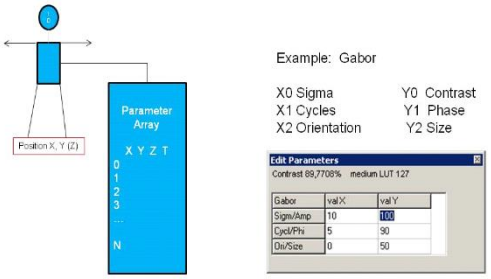


Figure 2: Variable Objects Fields (Suitcase)

Some Objects point onto other Objects using their ID (Figure 3). This is used for animation purposes or to organize a timely presentation (Animator, Mover, Shower). Colours are managed through Palette Objects.

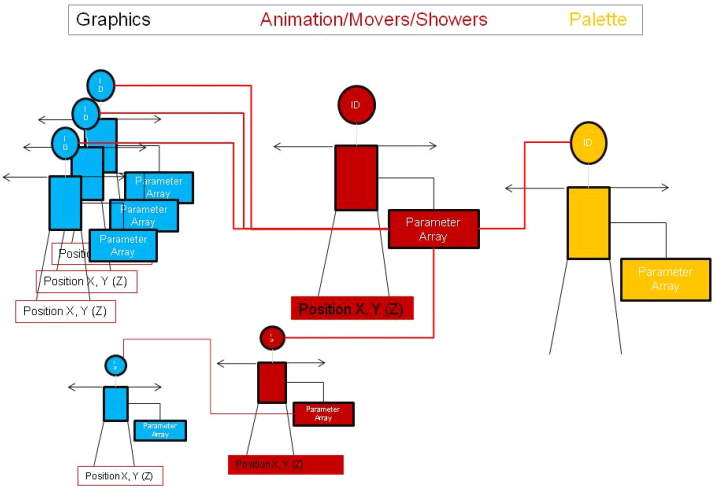


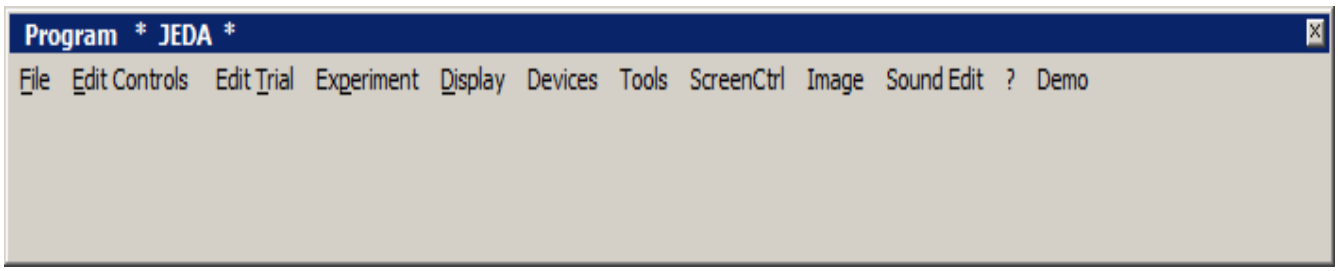
Figure 3: General Object types and pointers

A quick start example (also see Video Tutorial)

1. Launch NewJeda.exe : the Main Menu and Edit Menu show on

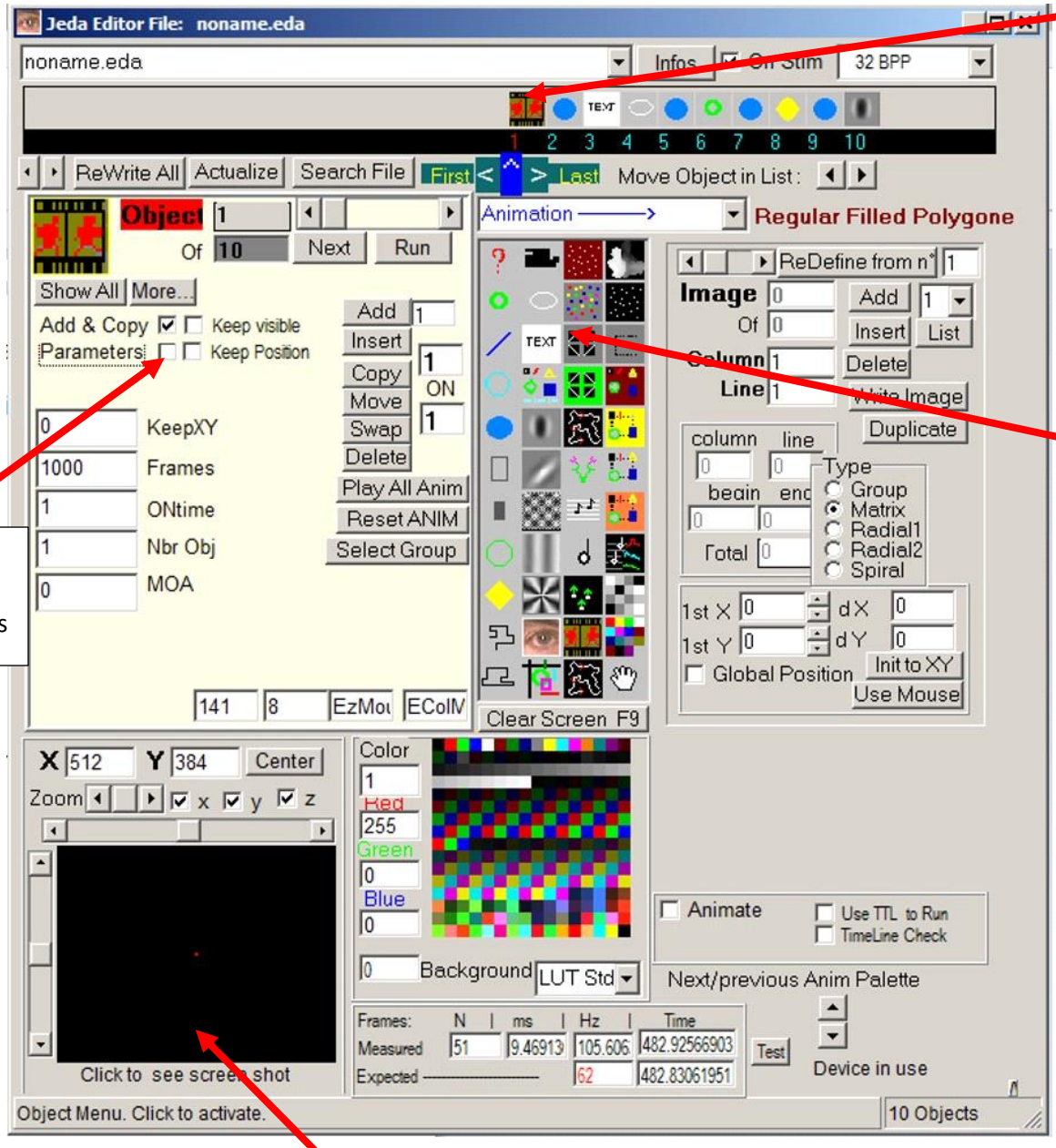
At start, Help Bubbles can be enabled/disabled in the **MAIN Menu`\Tools. Hints** are shown in the bottom Status Bar of the Edit Menu to gain information on the use of objects and functions

Main Menu



JEDA Main Menu

JEDA EDIT Menu



Object List

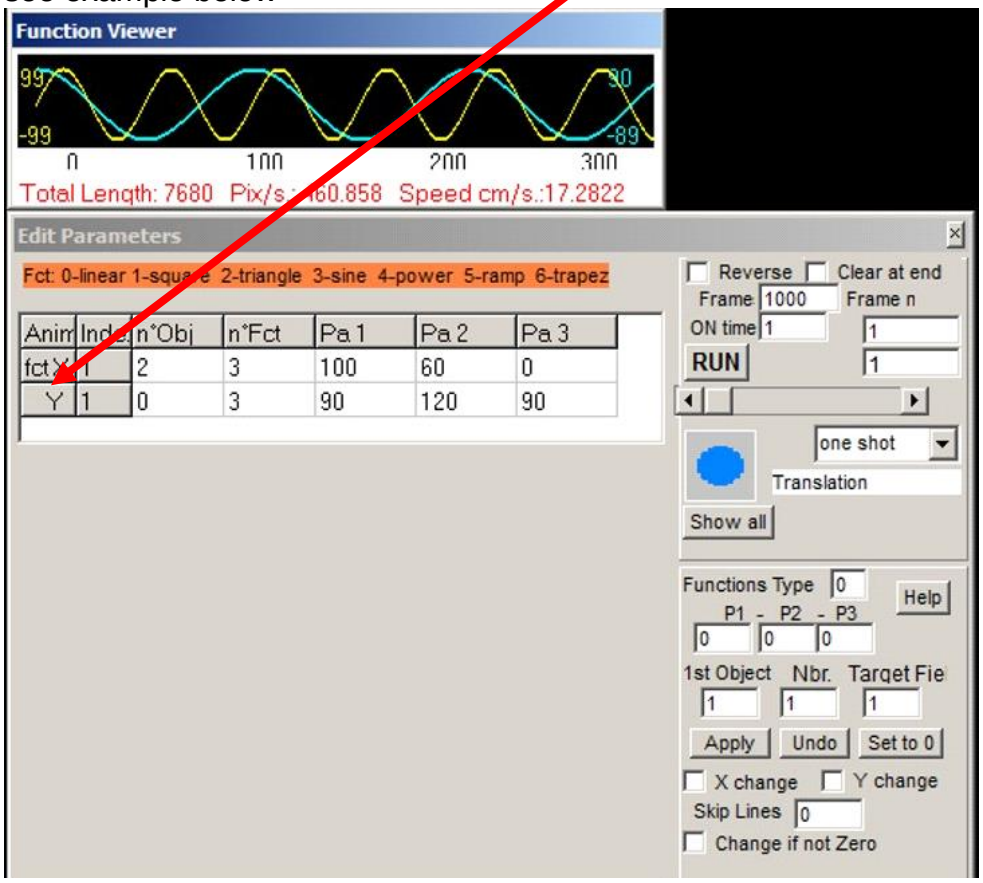
Icon Panel

Check to see Object Parameters

Edit Object Menu

Edit Screen

2. On the **Icon Panel**, double click on an Icon (e.g. a Disk) to edit a first Object -or drag and drop the Icon on the small Edit Screen.
3. Change the colour by clicking on the colour Palette, modify the width and radius
4. Choose/modify the Object position:
    a. By using the mouse on the small Edit Screen
    b. By entering values in the X and Y fields
    c. By using the ScrollBars of the Edit Screen
4. Add objects with the Add button (double-clicking on the Icon Panel replaces the current Object with the new one). If the 'Add & Copy' checkbox is checked the last edited Object will be duplicated otherwise a novel 'Unknown Object will be added.
5. To animate an Object create an Animation Object: Click 'Insert', and double-click on the Animation Icon 
6. Activate the 'Parameters Checkbox' and enter the ID of the Object to animate, the index of the function to use (for horizontal and vertical motions) and the parameters for these functions : see example below



Parameter Panel Array for an Animation

7. Click Run to launch the Animation (ESC to stop it).
8. You can add Objects in the Parameter Animation Window and set different motion trajectories. To add object Right click and select Insert New line or enter a number if the field 'Nbr Object' on the Object Panel. The ID of the Objects to be moved must be that of valid existing Objects.
9. To see an animated Object, maintain a Down Click on its ID. To go to the animated Object (e.g. to modify its parameters), Double Click on that Object: The current Object is shown, and a 'Go Back To Animation' Button appears. Click on it to go back to the current

Animation Object. To learn more on the functions one can use click on the ‘Help’ Button in the Parameter Pane (Also see the Function help in this document).
10. Load Examples to see different options and type of stimuli.

********** JEDA SHORTCUTS **********

F1 Contextual Help (current Object Type)
F2 Launch animation if compatible with Object Type
F3 Access to Device Control
F4 Access to Colour Control
F5 Next Object
F6 Previous Object
F7 Access to Copy Menu
F9 Clear Screen
F12 Change current File to next File.

CRTL O: Open File
CTRL S: Save File
CTRL L: Close File
CTRL N: New File
CRTL B: Save Objects as BITMAPS
CTRL Z Cancel last operation
enabled after modifying an Object or Image
CTRL Y: Redo last Operation
Right Click: Contextual Menu

JEDA HELP-general

Objects in Jeda

(Use CTRL + Click to access object Help)

OBJECT LIST CONTROL

Object Selection: click on Object Icon in the Object Panel

When Object or Icon is selected:
CTRL X Delete Object
CTRL C Copy Object
CTRL V Paste Object
CTRL I Insert Object
The **Next Button** search for the same Object Type in Object List (F5)

Changing and validating the value of a selected Object's field:

Enter new value in field
TAB key or CR
TAB: next field
Shift TAB: previous field
F5 -> Next Object (same field is selected, allowing for quick changes of a field of several Objects)
F6 -> Previous Object (same field is selected)
Shift + Field selection with mouse Click: Moving Mouse up and Down changes Field value

Keeping Objects Visible on Screen

Click on **KeepVisible** or shift-click on List Elevator
-> Enlarged width elevator indicates enabled KeepVisible Function
-> Reduced elevator indicates disabled KeepVisible function)

Object Position in Object List can be modified

by dragging and dropping the '>' sign beneath Object Icon
'Copy' 'Swap' and 'Move' Buttons can also be used

* Setting OBJECT POSITION on Screen *

1. Click on Object Icon
   CTRL Arrows moves Object
   Arrows Change the current Object in the LIST
2. Select Object in the "local" screen (Current Object is shown by a red dot)
   Use Mouse to change Object Position
   Alt + Mouse + right or Left Click to expand or contract Object
   Shift + Mouse + right or Left Click to rotate Object
   Ctrl + Mouse for fine position control
3. Set the X and Y coordinates directly in the X and Y Edit boxes

* SCREEN CONTROL *

F2 Launch animation if compatible with Object Type
F9 Clear Screen
Screen parameters available in the CONTROL Panel "Screen Control"

*CHOOSING OBJECTS *
Available Objects are set using the "Object Scrolling List"
`See example AllObjects.eda File in Jeda\DemoJeda directory`
**** GRAPHICS Objects summary (see below for a detailed description) ****
Dot: 4 parameters
Line: 5 parameters
Circle: 5 parameters
Disk: 5 parameters
Empty rectangle: 6 parameters
Filled rectangle: 6 parameters

For the following Objects: --->Mouse Edit available with the "More..." Button
(Note: Left click to select, Right click to unselect; ESC to quite Mouse editing mode)
- Empty regular polygon: 6 + n parameters
- Filled regular polygone: 6 + n parameters
- Polyline: 5 + n parameters
- Empty polygon
- Filled polygon
- Empty Ellipse
- Filled Ellipse
- Text: Enter Text in the Text Edit Box
- For the following Objects define the Palette => Set a Palette with a linear (Gamma) function to use grey levels
- 2D Gabor
- Gabor
- Spiral
- Plaid => Set a Palette with a sine function and use a Palette Rotation to animate
- Grating => Set a Palette with a sine function and use a Palette Rotation to animate

Bitmap (*.bmp): To load from disk use TAB key in the File Name Field (Several Bitmaps can be loaded at once)
Dot XYZ: the More Button … provides direct Mouse Editing
Optional: Load a XYZ file from disk. See below for details
Fractal

* SOUND Objects *
- Pause & Sound (uses the PC speaker)
- Sound (Wave *.wav File)
- To load from Disk use TAB key in the File Name Field (Multiple Sounds can be loaded at once)

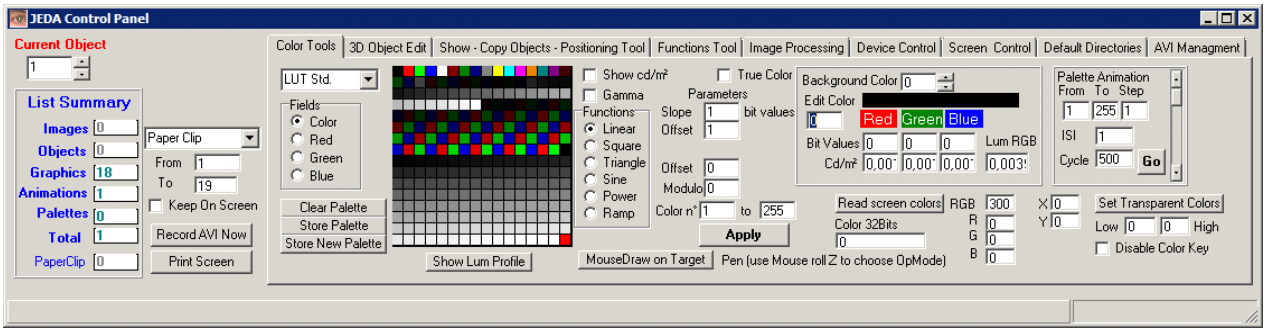
* POINTER Objects *
Pointer Objects use Object's index to specify which Object to Display
Pointer Objects have a "-->" sign in the scrolling list: Index is the ID number of each Object shown in the Object Panel

Group-> shows one or several groups of Objects
Matrix --> creates a Matrix from one object
Mask --> creates mask (set transparent color, as a negative number)
Mask XY --> as Mask, but objects have XY coordinates)

Object Animation --> use functions to translate, rotate and expand objects (see help for details)
3D Animation --> Move 3D objects
Random Dots --> Creates RDK : optional use a target Object; (See contextual Help for details)
Group Show --> Shows objects in a specific time sequence
List Show --> Shows sequences
Ima Show --> Shows images in sequence
MoveAnim-->Use XYZ arbitrary values to animate objects the '(More' Button enable direct Mouse Editing on Display)

Jeda's CONTROL PANEL

Jeda's Control Panels gives access to a number of parameters and tools to control: Colours and palette, 3D objects, Show-copy-Positioning Tools, Function Tools, Image Processing, Device Control, Screen Control, Default settings, Movie editing



******* LIST SUMMARY *******

Provides an Overview of the number of Objects and their Type in the current List

Also see the **Show List** Button in the IMAGE Control to see the Image list in details

******* PAPER CLIP *******

Use the paper clip to Copy, Paste, Show, Cut, Add, Clear Objects from #n to #m (A sub-list of Objects defined by a lower and upper index ID)

******* COLOR TOOLS *********

or (LUT: Look Up Table)

Function Palette: defines a Palette using functions

Raw Palette: stores a specific palette (use the **Palette** tool)

Palette Animation: dynamic palette

See Control Panel "Color Control"

******* FUNCTIONS ******

JEDA uses functions to assign values to fields:

1. Of an Object list
2. Of an Object (trajectory for a Mover and DotXYZT)
3. Of a Palette

See Help or Control Panel "Function Tool" or Functions Parameters below

******* 3D OBJECT EDIT ******

This tools provides functions to work on 3D Objects (Dot XYZT Object, see below)

Loading predefined 3D Object (*.xyz, *.wrl) is available from the OBJECT PANEL

Hit Tab KEY in the File Name Field of Object Panel

Edit

Generate

Move (Rotation, Zoom),

Modify 3D clouds, etc.

**** IMAGE PROCESSING ******

This tools provides functions to modify Objects using a DirectX Surface (a bitmap)

Saving a snapshot of Display Screen

Split a previously loaded Bitmap in smaller bitmaps

Combine several Bitmaps

Visualize the FFT spectrum of a Bitmap

Inverse FFT to generate Images

Combining Power and phase

Morphing Spectrum of several Images

Filter a Spectrum

**** DEVICE CONTROL ******

Device control and checking configuration for:

Keyboard

Mouse

Joystick (including Force Feedback)

Tablet (Wacom)

Eye Movements (Eye Link II: also see the Experiment Menu)

Also see Devices in Main Menu

**** SCREEN CONTROL ******

Set Display Configuration

Check Refresh Rate and timing

Visualize Calibration Settings (values set in the ConfigJEDA.cfg File)
Linearized Palette (Gamma corrected) uses the following function for the Red Green and Blue guns:

```
L = offsetBlack+ Slope * pow(BIT, expoV)  expressed in cd/m²
```

Calibration and fitting to this function should be done to provide valid Luminance Values
Indicated in the Colour Tool (see Palette Below)

FIXATION point can be set: Size, Colour, Type, Position
Define a type: from 1 to 7:1: Cross 2: Four dots 3: Rectangle 4: Circle 5: Cross 6: Filled Square 7: Disk

A grid can be enabled to provide a reference frame

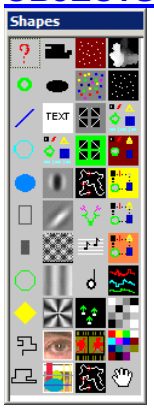
**** DEFAULTS DIRECTORIES ****

Indicates the default Directory for JEDA
Default Directory can be loaded, changed and Saved in the ConfigJEDA.cfg File

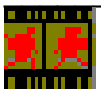
Bubble Help (appearing when a Control is under the Mouse) can be enabled
To activate the Help Bubbles go in the 'Tools' Menu. A Hint will be displayed for each Control

**** AVI Management ****

Saving a Film (AVI Format) can be enabled by checking the "Record AVI" box, and then running an Animation. The Film will store the frames of an Animation or all the frames changes by the user.
It is also possible to load and play Movies of different formats (depending on the compressor that are installed. Install ffdshow codecs on your computer.



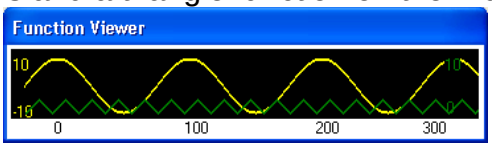
Double Click on an Icon of the Icon Panel to set the current Object type, or drag and drop an Icon of the Mini Control Screen to add an Object to the Object List

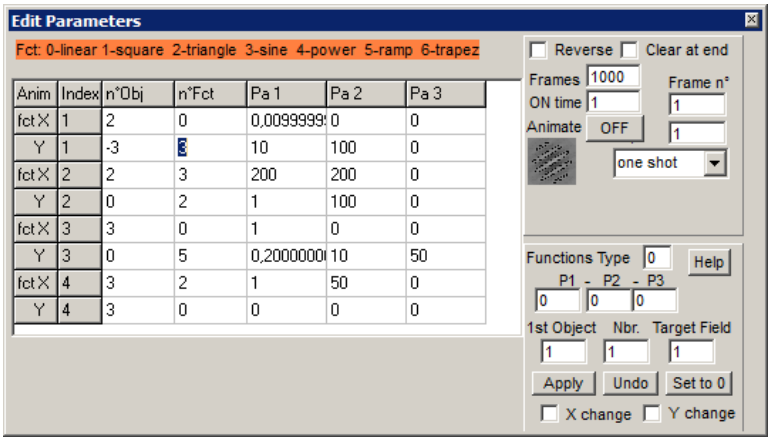


Anim Object

The Anim Object is a powerful tool providing a number of options to present, move and transform other objects, including other animations (3DAnim, Random Dot Kinematogramms, Groups, including the Anim Object itself).
The number of Objects to be moved is indicated in the Object Panel and can be changed at will.
The Parameter Array (below) provides access to the values to be set to choose the desired animation. Options are available through the Help button or a right click.
Each animation is defined by two Set of parameters referred to as the X and Y parameters. Each line comprised 5 fields. The Object field X refer to the Object Target to be animated.
Several Animations with the same object can be defined in a row with the consequence that the different Animations will be added.
The Example below indicates that Object #3 (a Gabor) will be rotated and expanded (Line 1/1) while moving along a sine on the X axis and a triangle function on the Y axis (line 2/2).

Function Window and Panel





The Function Window displays the X and Y function currently defined.
Moving the Animation itself allows moving all objects as a whole.

Complex motion

Several Animations with the same object can be defined in a row with the consequence that the different Animations will be added.
See Example in JedaHELP.doc

The Function Window displays the X and Y function currently defined.
Moving the Animation itself allows moving all objects as a whole.

Checking the timing of an animation:

Select the object to verify and hit F2. At the end of the Animation a Window displays all frames, timing and positions over time of the Object.

Morphing Animations

A series of Animations can be morphed to smoothly change the trajectories and motion type. This behavior requires that several similar Animations are defined in the Object List. The KeepXY filed in the Object Panel indicates the number of Animations to morph and the Morphing Period then allows setting the duration of a Morph.
Setting KeepXY to 1 serves to use the last positions of a previous Animation as the initial position of the next Animation.

Changing the parameters of several animations at once

In the Function PANEL (on the right):

1. Enter the function type, and set the P1, P2, P3 parameters
2. Enter the lower Index and the number of Values to change
3. Select the column (Target field) on which the function should apply
   (-1, 0, 1, 2 or 3)
4. Check the X_change and/or Y_change CheckBox, depending on whether you want to modify X and/or Y values. (Using Up_Down arrows changes all values of a row)
5. Click Apply or Undo buttons to change the selected values;
   Set to Null to reset the selected values to 0.
6. Checking X and/or Y check box, select a column and use the up/down arrows
7. Using the Ctrl, Alt or Shift key together with the up/down arrows

---

FUNCTIONS CODES AND PARAMETERS

As it is often useful to set the parameters of many Objects in JEDA, using mathematical Functions, Y=F(x) is a very general and convenient way of quickly modifying the fields of many Objects at once: this is used for Palette colors, Animations, Field of a group of Objects, coordinates of DOT Objects, etc. Each function has 3 available parameters, P1, P2, P3.

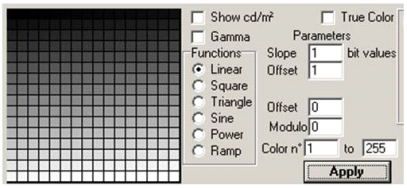
The principle is to select a function (defined by a number), to set the desired parameters values for that function, and to apply the result to the selected parameter.
For instance, one may select a Linear Function Y=Ax + B, set the values of the slope A and offset B, and to use the Result Y to set the RGB values of a Palette. Here, x refers to the indices of colors.
Another example is to use Functions (or combination of Functions) to move an Animated Object (as in the example described here). In that case, x is the frame number. When one wants to change the parameter of a series of Objects, x is the number of Objects.

Using Functions in JEDA (see HelpFunctions for a list of functions)

Functions can be used to change
: a list of Objects, Objects' positions or parameters
: change Colors,
: Animate Objects

EXAMPLE : Y = ax + B linear function
x can be a : color index; object index; parameter index

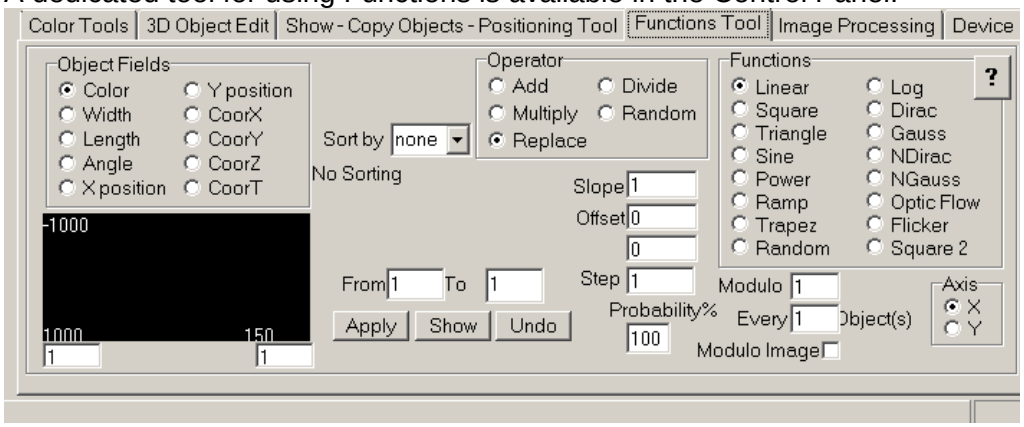
Color example : apply linear function y=ax + b to color indices from 1 to 256
With a=slope=1 b= Offset=1



color 1= a x 1 +b
color 2= a x 2 +b
color 3= a x 3 +b
…

In an Animation, Positive Function codes in X field #P0 indicate the Objects' Indices
Negative Function codes in Y field #P-1 (below the Object Index) define specific behaviors (rotation, expansion, color changes, morphing, etc.). By default, 0 indicate a change in Object's position.
Use the Popup Menu (Right Click) to insert a Function code.

A dedicated tool for using Functions is available in the Control Panel:



To use this tool:

1. Select the field to be modified : Object Fields
2. Select the Operator to use: Adding, replacing, etc. (for instance selecting Adding will add the results of the Function, Y, to the current values of the selected Objects' field)
3. Choose the Function to use: Functions
4. Enter the values for each parameter
5. Select the limits (First and Last Object to modify)
6. Apply the Function (Undo is available)

For instance, if many Disks of the same color have been edited, it is possible to change their color or size at once with a function, or to modify a specific field parameter of a series of Objects (e.g. the orientation of Gabors).

The Codes (number) and type of Function is listed below

0 Linear Y = Ax + B
P1 is the slope A
P2 is B
P3 is used to produce a step function
(Note: use this function with A=0 to add or subtract a fixed value B or to show a static Object)

1 Square :

P1 is the Amplitude A of the square function
P2 is the Period T
P3 is the Phase PHI

2 Triangle
Uses two linear functions (Ax+B) with alterning slopes:
P1 is the Slope
P3 is the Offset value
P3 is the Period of alternation

3 Sine Y=A*sin(2pi/T + Phi)
P1 is the Amplitude A
P2 is the Period T
P3 is the Phase Phi

4 Power Y= pow(x, A) + B)
P1 is exponent A
P2 is the offset B
P3 unused

5 Ramp
Linear Function with periodicity
P1 is the slope A
P2 is the ordinate B
P3 is the Period T

6 Trapezoidal
P1 amplitude
P2 Period
P3 0ffset

7 Random
Sets a random variable Y=+- random(B)
P1 is the life time, in frames, of Y value
P2 is the value for randomization B
P3 is the sign of the random number:
(0 or 1 gives positive numbers,
>2 gives negative And positive numbers
<0 gives a single negative or positive value
for the whole duration of the Animation

8 Log Y=(LOG(X*x+A+B)

9 Dirac Single pulse :
P1 is the time offset
P2 is the pulse amplitude
P3 is the pulse duration

10 Gaussian
P1 is Sigma
P2 is the Amplitude
P3 is used to center the x variable

11 Dirac Multiple pulses :
P1 is the time offset
P2 is the pulse amplitude
P3 is the pulse duration

12 Multiple Gaussians
P1 is Sigma
P2 is the Amplitude
P3 is used to center the x variable

13 Optic Flow
P1 is the position relative to center
P2 is the center position
P3 is tau, the expansion/contraction rate

14 Flicker
P1 unused
P2 is the pulsation
P3 is the phase

15 Square Uses the sign of a sine function to build a Square function.
Sign (sine(2PI/T+Phi))
P1 is the Amplitude A of the square function
P2 is the Period T
P3 is the Phase PHI

SPECIAL CODES below Object Index (field Y) are specific to Animations

---

N° ID to chain motion of different objects: an Object inherits from the motion of the Object ID if the index ID is in the list of animated Object (Warning ! Not the index of the Object itself).

-1 Rotation (defined as X function)
-2 Expansion (defined as Y function)
-3 Expansion & Rotation (cf. above)
-4 Color LUT index (ranges defined line X)
-5 Luminance of specific color index defined on line Y, P1; P2 defines a Bit or Luminance change (0 or 1); P3 allows to modify specifically RGB channels
-7 Morph Objects :
Xfields: P0: 1st Ob#1, P1 last Ob#2, P2: Cycle Duration P3: Morph Type (1 to 6 types available)
Yfields: Threshold P1 used to finish morphing; P2 start with Time Offset in Frames
-8 Group rotation:
Xfields: P0:Ob1 P1:Ob2 P2:teta Dteta Slow
Yfields: - - P2 CenterX P3 CenterY -
-9 Mouse motion added to Object motion
if P1 (X or Y) is not zero motion along that axis is not enabled
if the last Object in List is a 'MoveAnim' or '3DotXYZ', mouse positions are stored in it.

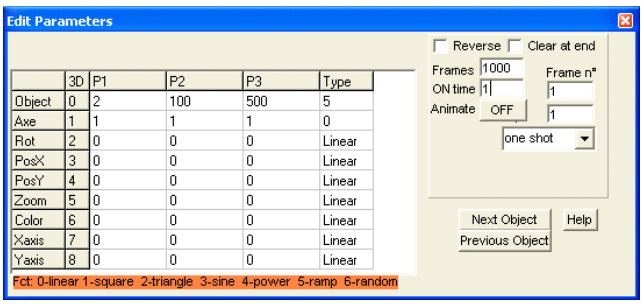
Dynamically Changing ANIMATION fields allow modifying the parameters of a Function on line
-10 : change P1 of the preceding Anim Line
-11: change P2 of the preceding Anim Line
-12: change P3 of the preceding Anim Line
-13: change P1 in cascade
-14: change P2 in cascade
-15: change P3 in cascade

-16 Use Eye Position (if EyeLinkII enabled)
-17 Time Functions: X and Y functions are summed to give the current time slice. It applies only to the next Animated object.
New Time functions must be define for new objects.

-20 Control Joystick with Force Feedback (on X and Y axis)
-21 Mouse control of Color or Sound Volume with Gaussian function
X fields : #P2 Amplitude #P3 Sigma
-22 Eye control of Color or Sound Volume with Gaussian function
X fields : #P2 Amplitude #P3 Sigma

Anim3D

An Anim3D Object is used to animate 3D objects: Rotation, Expansion, contraction can be applied to polygons, DotXYZ.
Several objects can be moved simultaneously, as defined in the Object Panel: Nb. Objects.
A full page of the Anim3D parameters is needed for each Object. The Next/Previous buttons are used to flip to the next or previous page.
The first Field in the Object Index
Different values can be set for each rotation Axis. Rotation on an axis is proportional to this value.
These values can be dynamically changes by functions using the Xaxis and Yaxis lines.
Different functions can be used to Rotate, translate, zoom an Object, as well as changing its colour. Enter function code or use the right click to set a function.

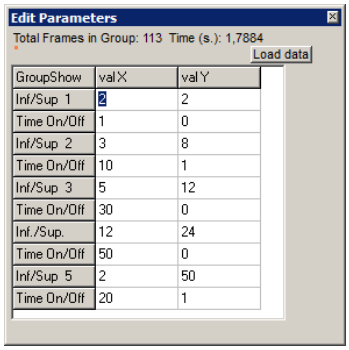


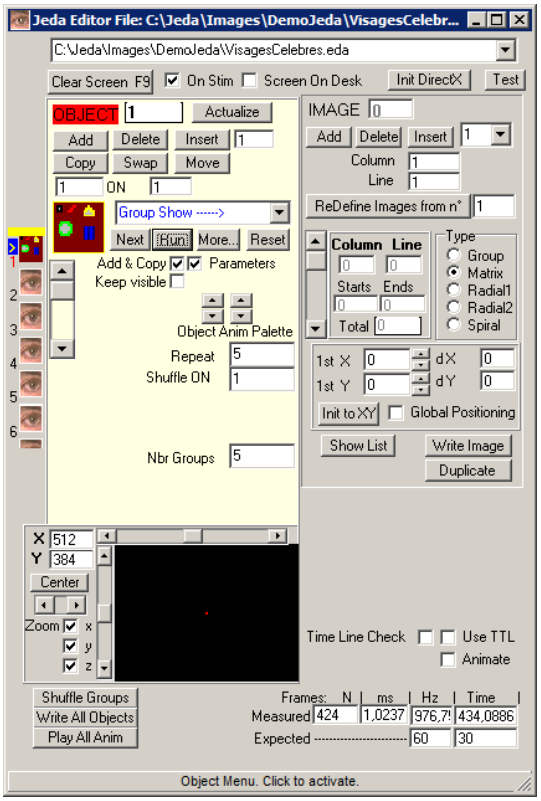
	3D	P1	P2	P3	Type
Object	0	2	100	500	5
Axe	1	1	1	1	0
Rot	2	0	0	0	Linear
PosX	3	0	0	0	Linear
PosY	4	0	0	0	Linear
Zoom	5	0	0	0	Linear
Color	6	0	0	0	Linear
Xaxis	7	0	0	0	Linear
Yaxis	8	0	0	0	Linear

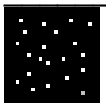
The P2, P3, Type value on first Line are used to run through a DotXYZ object using a Life Time. In the example above 500 dots will be displayed during 5 frames before moving by 100 dots in the dot Cloud so as to display the dots 100 to 600, ect... .

Group Show

A group show is used to display a sequence of events characterized by Duration and a time offset.
The sequence can be player several times: set the number of repetitions in Object Panel.
Upper and lower Indices of the Objects to be displayed at once must be defined.
Special code can be used to set the behavior of a Group Object : shuffling the order of the events or use a random presentation.
A Group Show can display Graphic Objects or Animations. The timing is that of the Anim Object in this latter case.



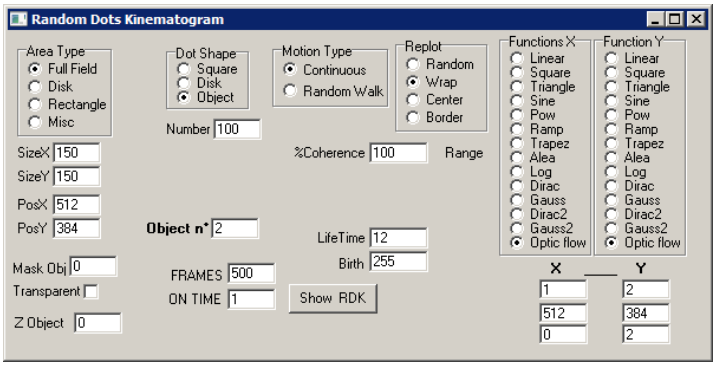




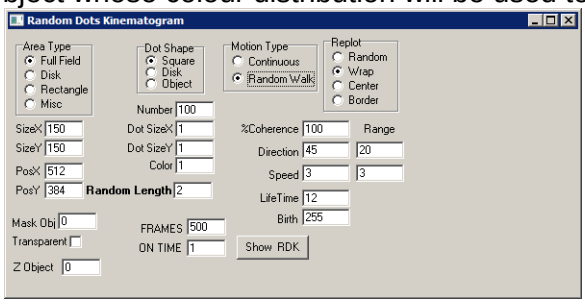
Random Dot Kinematograms (RDK Objects)

Random Dot Kinematogramms (**RDKs**) are used to define and move a cloud of objects.
When selecting a RDK Object, a panel with different fields is used to define the characteristics of the RDK Object.
Area Type defines the form of the region in which the dots are displayed. Define SizeX, SizeY, PosX & PosY to adjust this region as needed.
Motion Type: Continuous or Random walk can be used. When the "**Continuous motion**" option is set, the trajectory of the motion is defined with the **Function X** & **FunctionY panel** (refer to "using functions" in Jeda).

The Random walk Option provides fields to define the *Direction* and its associate distribution (*Range*: direction will be chosen at random for each “dot”), the *Speed* (in pixels/frame) and its associate distribution (Range, speed chosen at random for each “dot”).
Life time defines the number of successive frames during which a dot will remain on screen; afterwards it will be randomly replotted in the Display Area depending on the Replot setting.
A Random Dot Object can be used -pointed by- in an Animation Object. Thus, it can be moved or dynamically modified. It is also possible to morph RDKs with different settings in an Animation Object. For instance, two RDKs with different speeds, directions, life times can be morphed. The parameters of the RDKs will be smoothly modified during an animation. Warning: th morphed RDKs must have similar motion settings.
An object used by a RDK can be modified during the RDK motion (e.g. morphing a RDK target object with another, or rotating a RDK target Object).



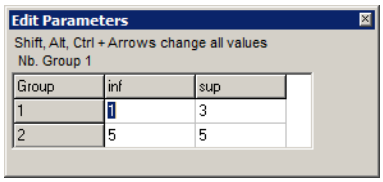
In Continuous mode Functions for X and Y motion can be set independently. In the example above an expanding Optic flow (centered at 512/384) moves in spiral motion.
Several RDKs can be moved simultaneously by an Animation Object (see Animation Help).
RDKs can move Graphic Objects or Group Objects. A group Object can display a sequence of different Graphics Objects (see Group Help) the identity of objects moved by a RDK can change over time.
Misc indicates that an Object of the list -e.g. a mask- will be used. Its index should then be defined in the Mask Obj. Control.
This mask is used to limit, or hide, a portion of the RDK area. If *Transparent* is checked, the mask should be a true *Mask Object* with transparent colour equal to target Objet colour (only drawing object with a single colour will work properly).
Dot Shape: the RDK can use Dots (with any colour & size) or any other Object defined in the Object List. In this later case the index of the Object must be set in Object n° field.
Number defines the number of objects in the RDK.
ZObject defines a target Object whose colour distribution will be used to modulate local dot speed.



Example of Random Walk with a square Dot Shape: Life time defines the time during which a dot is moved before being replotted. Birth Time is the time of birth of each dot (randomly chosen between 0 and 255 in the example above. Direction is 45° + or – 20°. Speed is 3 pixel/frame + or - 3

Group Object

A Group Object displays several groups of objects defined by *a lower and upper index (Objects' ID)*.
Several modes, defined in the Object field n° 2 ("All Objects") in Object Panel, are available:
By default the mode is 0 and correspond to "Show all objects" where all defined objects are shown simultaneously.

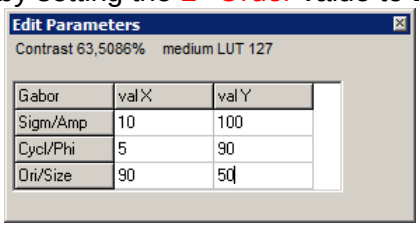


Other modes are used when a Group Object is displayed within an Animation:
A Group Object displays several groups of objects defined by *a lower and upper index*.
Several modes, defined in the Object field n° 2 in Object Panel, are available:
By default the mode is 0 and correspond to "Show all objects" where all defined objects are shown simultaneously.
Other modes are used when a Group Object is displayed within an Animation:
-1: Randomize subgroups once at Animation start
-2: Shows one or several Object(s) chosen at random.
-5: Shows objects in sequence, one at a time for each group
-6: Same as -5 but in inverse order
Except with mode -1, a Renew Rate (or Life Time in number of frames) must be defined to indicate the rate at which a new choice of Object should be made. A Time Offset can be defined to set the initial onset of a display (sequence start). Life time or renew rate must be defined. A time Offset can be defined to set when the sequence starts
Positive numbers are used to display M Objects amongst N (greater than M) Objects at random
A Group Object can be rotated around its centre (Objects of the GroupObject are not rotated, only their XY position changes)
If an AJU File is stored during an Experiment, the index of the group and the indices of the Objects displayed by the GroupObject are written in the AJU file
If the MEG-EEG flag is set, a TTL is send onto the parallel port (See Markers).

Gabor Objects

Gabor Objects are sine wave gratings modulated by a Gaussian profile. The parameters of the Gabor are available in the Parameter Array.
With simple 1DGabor (See below):
Sigma is the width of the Gaussian window defined in pixels. Note that with large value a Gabor will look like a sine wave grating.
Amplitude is the modulation of the sine wave defined in color indices of the current Palette. This means that the pixels are drawn from the current color Palette using the upper and lower color indices defined in the Object Panel. A Gabor can use the whole Palette or only a subset of entries (as defined in the Object Panel). Depending of the Palette settings colored or grey level can be used. The Palette (or LUT : Look Up Table) can be changed at will in the Control Menus (access with the F4 key).
Cycle is the number of the cycles of the sine wave function given the size of the Gabor (it is not the spatial frequency!)
Phase is the phase of the sine wave grating.
Orientation is the orientation of the sine wave gratings

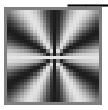
Size is the size in pixels of the surface used to draw the Gabor. When the size is large relative to sigma the Gabor is truncated to 2 sigmas.
A second Order Gabor can be drawn by setting the 2ndOrder value to 1 in the Object Panel.



Animation of Gabors :
Gabor can be dynamically changed either by

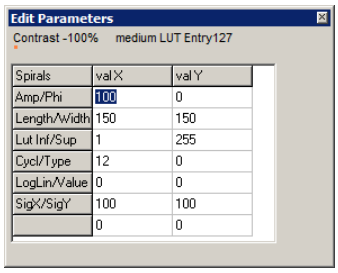
1. Morphing two Gabors with different parameters (Use an Animation Object to set the speed and type of Morphing. Morphing can be tested in the Control Panel: Function Tool).
2. Rotating or expanding/Contracting a Gabor can be done within an Animation Object (see Help section of Animations)

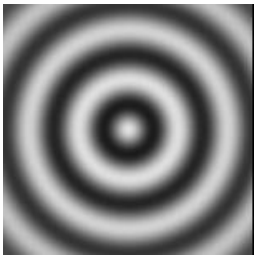
A 2DGabor Object is similar to 1DGabor except that both the X and Y sigma and Size can be set independently (see below).



Spirals

A Spiral Object is a graphic object for drawing a variety of concentric, radial and 2D patterns
It is defined by:
- An amplitude modulation and a phase value. Amp refers to the contrast (in fact the extreme Entries used in the Palette). Its value should be set with regards to the Lut Inf/Suf values. Phi refers to the phase of a spiral
- Length/Width define the size of the spiral bitmap. The XY size is convolved with a Gaussian window (Sigma X and Y)
- Color entries (e.g.from 1 to 255 for grey level palettes). Lut Inf/Sup define the lower and higher entries of the palette to be used. Using only a subset of the available entries allows that different spirals are drawn with indeopendent colors (with the cost of reducing thy number of entries for each).
- Cycl/Type: Cycl indicate the number of cycles in a spiral.
: Type, associated with LogLin/Value, defines the behavior as a function of type.





Available Types are:
0: Concentric
1: Radial
2: Spiral
3: Checkerboards
4: Sectors
5: Xor
6: Biased
7: Or
8: Checkerboard

Changing the Palette (in the Color Menu) modifies the aspect of a spiral.

In the main Object Panel, the field Second-Order, when set to 1, draws a second-order spiral. Higher values limit the size of the spiral.

See the file `SpiralExamples.eda` in the Jeda/Images/DemoJeda directory

Bitmap (*.bmp files only)

A Bitmap is loaded by entering the Tab Key when prompted in the File Name control of the Object Panel
A Bitmap is stored as a FileName pointing to a bmp File
Bitmap size is managed through X and Y zoom
Zoom values are stored in the parameter array as a tab[0].z/10 and tab[0].t/10 numbers (these can be modified directly or through the "Function Tool" of the "Control" Panel).
Dynamically Expanding/Contracting a Bitmap can be done with an Animation Object (see Anim Help)

In 8 bit mode the palette can be modified to set the Bitmap colors at will.
Note that bitmap color is different if the palette or the Bit Depth (BPP) in the control Panel are changed *before* or *after* loading a Bitmap
A bitmap can be combined with other bitmaps referred to as indices ranging from Bmp Index Inf to Bmp Index Sup
Logical operators can be used to manage Bitmap Combination
Logical operators are as follows:
0: SrcCopy;
1: Blackness;
2: SrcInvert;
3: NotSrcErase;
4: NotSrcCopy;
5: MergeCopy;
6: SrcErase;
7: SrcPaint;
8: DstInvert;
9: MergePaint;
10: PatInvert;
11: PatPaint;
12: SrcAnd;
13: SrcInvert;
14: SrcPaint; 15: Whiteness;

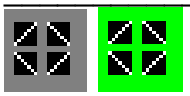
Mix Objects

A MixObject combines the bitmaps (or surfaces) of Graphics Objects according to a mode (between 0 and 16). Several couples of objects can be defined (Nb.Mix).
Color defines the MixObject background colour

Parameter Panel:
Objects to mix must be specified by their indices in the Parameter Panel.
The X and Y size of the MixObject bitmap is specified on line 1
X and Y values are used to defines offsets relative to XY Mix size

Mix Objects can be used to define Plaids or sum or Gabors or any combination of Graphics Objects.
When several couples of objects are defined the colours are add in pairs using
One of the following modes:
0: Mix Color+= Color1 x Color2/MAXCOL
1: Mix Color+= sqrt(Color1Color1+Color2Color2)
2: Mix Color+= Color1-Color2

3: if (Color2) Mix Color+= Color1/Color2
4: Mix Color+= Color1 ^ Color2
5: Mix Color+= Color1 & Color2
6: Mix Color+= Color1 | Color2
7: Mix Color+= ~Color2+ ~Color1/2
8: Mix Color+=(Color2>Color1) ? Color2 : Color1
9: Mix Color+=(Color2<Color1) ? Color2 : Color1
10: Mix Color+= Color1<<1
11: Mix Color+= Color1>>1
12: Mix Color += Color1
13: Mix Color += Color2
14: Mix Color+= (Color1+Color2)%MAXCOL
15: Mix Color+= sqrt(Color1 x Color2)
16: Mix Color+= (Color1+Color2)/2



Mask Objects

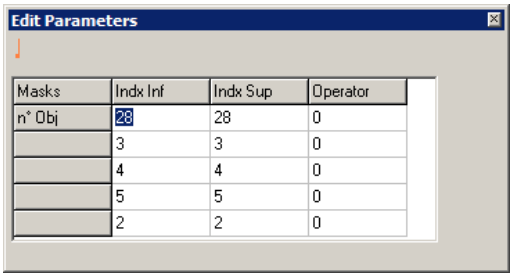
Mask and Mask XY:
A Mask is an Object using other objects to make up a complex stimulus
It is used to:

1. Make a transparent hole in the Mask surface. To this aim, a target object used to define the shape of the hole must be defined, and its color index entered as a **negative** value so as to specify which color is transparent.
   Several mask Objects can be used (accessible through the Parameter Window)
   - Objects with continuous indices are entered as a group defined by the low and high indices, eg from 1, 5.

- Several groups of mask objects can be defined in different lines of the paramters window

2. Combine graphics objects

Mask Objects can be combined with logical operator (the Op field in the Parameter Window)
to add, subtract, etc. target Objects



Logical operators are:
0: Ignored
1: Add
2: Substract
3: Divide
4: Replace
5: Xor
6: And
7: Or
8: Complement
9: Left bit shift
10: Right bit shift
11: Binarize

The MaskXY Object has similar properties as Masks but treats objects one by one to specify the Target Object XY positions within the Mask.
By default a Mask occupies a full 1024x768 screen and has the 0 background color entry
Size and Color and be changes at will.
(Warning, the Target bitmaps must be included within the area of the Mask)

DotXYZ object

Editing a DotXYZ object

A. A DotXYZ object can be loaded from a file or created with Jeda editing functions.
Press the TAB key in the File Name field to access stored files. If loaded from disk, a file should have the following format:

```
N    (= Number of dots integer or long integer)
X1   Y1   Z1
...  ...  ...
Xn  Yn   Zn
```

Where X, Y, Z are floating point numbers written as jjj.iiii separated by a blank. After loading the file name will be replaced by “OK”

Warning: to edit an Object from scratch enter OK as a File Name

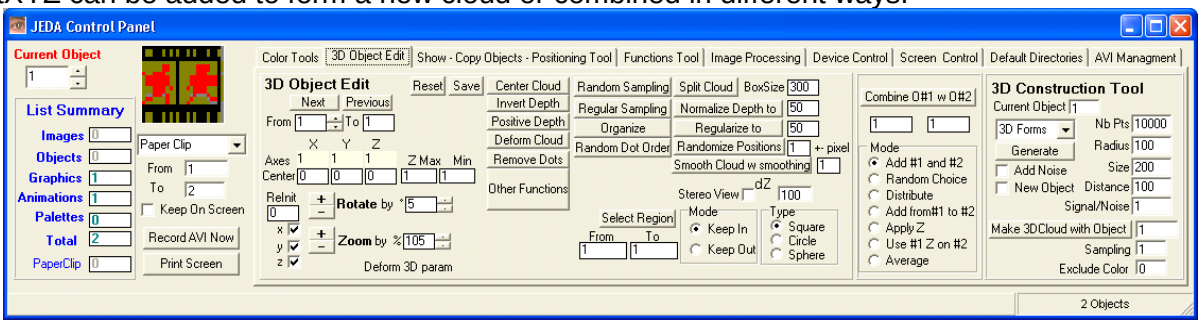
B. A DotXYZ object can be edited with the mouse or the tablet device (in this case, the pressure will be used to set the Z values). Draw with the mouse and hit Escape to quit. Note that moving the mouse with the right click pressed erases the dots.

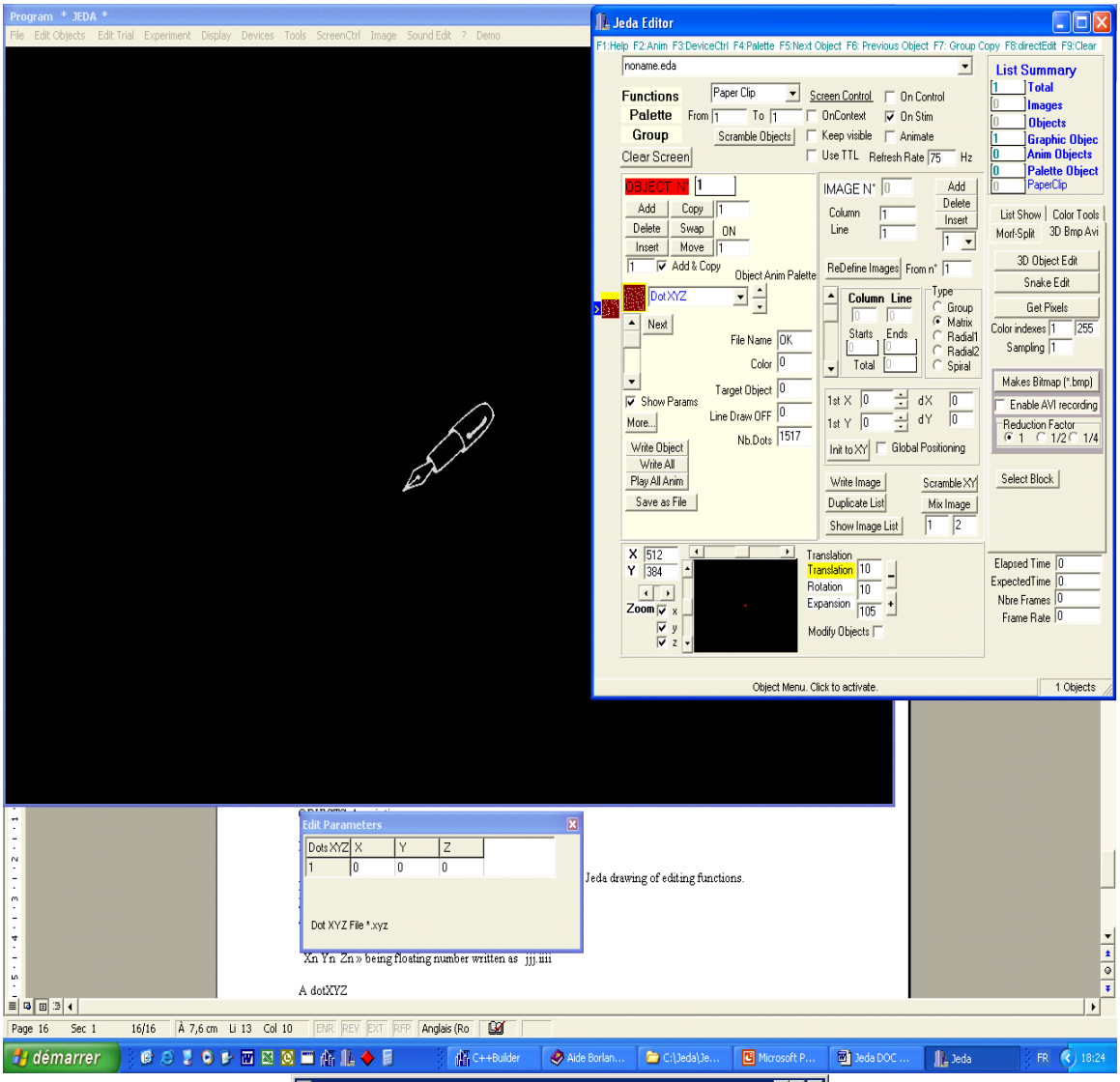
C. A DotXYZ object can also be edited with the “Function” tool. First specify the number of dots in the cloud. Open the “Function Tool” menu in the Control Panel. Check the CoorX, CoorY or CoorZ and enter the lower and higher indexes of the dots you plan to modify when applying a function. Choose the function and parameters for that function. Select the Object(s) on which this function should be applied and click “Apply”.

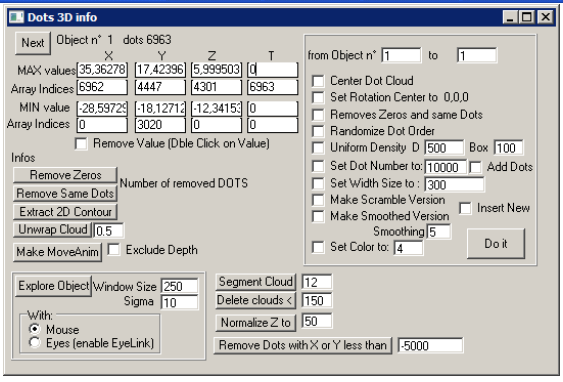
D. A DotXYZ can also be created from the bitmap of an existing Object (e.g. a Gabor patch or a picture). Open the 3DObject Edit tool in the Control Panel. Enter the index of the target object to use. Click on “Create 3D cloud from Object n° #“

E. Predefined 3D Objects are available in the 3D Construction panel.

DotXYZ can be added to form a new cloud or combined in different ways.







2. Modifying the appearance of a DotXYZ object:

A DotXYZ can have a specific colour from 1 to 255. If Colour is set to zero the Z value will be mapped onto the palette. Thus dot colour will change with Z value.

If Colour is –1, -2, –3 only ¼ of the dots, chosen at random, will be drawn.

A “Target Object” can be used to draw the cloud. This value must be the index of another valid object in the object list such as a disk.

The Line Draw field can be used to trace lines in between dots. The value will define the maximun length to be traced.

If a Target Object is defined the “Line Draw” value will be used to draw the Target Object with a size proportional to the Z value.

The number of dots in a cloud can be changed at any time. Enter the new value.

The zoom can be used to enlarge the could on whatever dimension (x,y,z), depending on checked item in check boxes.

These check boxes are also used to load a single dimension (X, Y, or Z) of a XYZ file. To (re)load a file, erase the “OK” characters of the “File Name” and enter Tab. An Open Dialog Menu will allow to choose a new XYZ file.

The 3D Object Tool can also be used to manipulate DotXYZ objects. Rotation, Expansion, Z normalization, selection of a subCloud, addition of Clouds and other functions are available in this menu.

The panel above accessed through the Other Functions in 3D object Edit can be used to perform series of transformation on a group of objects.

Animation of DotXYZ cloud is performed with the 3DAnimation Object

---

PALETTE OBJECT

By default, JEDA uses a palette of 256 colour entries, each defined by a Red-Green-Blue triplet of values ranging from 0 to 255. The Colour Table is accessible in the Edit Menu or through the Color Tools in the Control Panel (hit F4, See below).

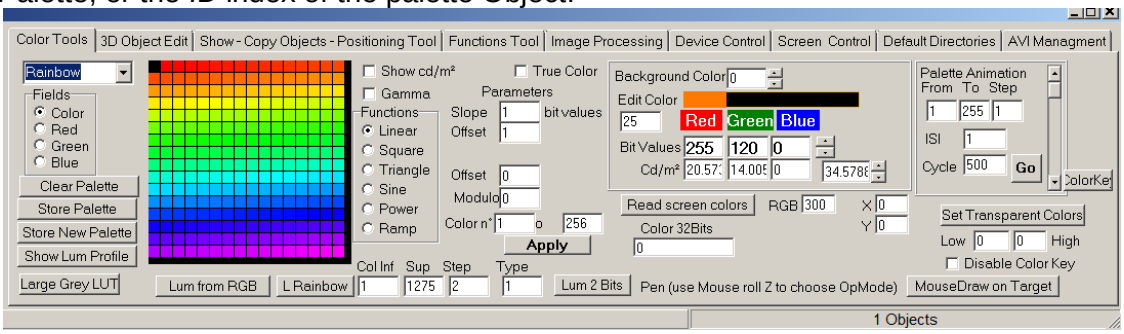
Colour tables can be changed using default Palettes or using Functions. The functions can apply on RGB values simultaneously (Grey) or independently on each of the R, G, B channel.

Colour Tables can be stored as Object in the Object List. A Palette Object contains an array of RGB values and can be edited by setting the RGB values for each entry in the Parameter Array.

To store the current PALETTE as a Raw Palette Object:

In the Control Panel-Color Tool click on Store Palette if the current Object is a Palette to be modified or click on Store New Palette if a new palette is needed. The lower and upper indices of the Stored Palette are those defined in the Color Tool.

In 32 bpp mode, some Objects (Gabors, Gratings, Spirls, Plaids) may use gray level palettes. These palettes can be added as Objects (and be further edited as needed) or are hidden. For these Objects, a field in the Parameter Arrays allows choosing the Option 0 for a hidden Grey Level Palette, or the ID index of the palette Object.



If the calibration parameters are correctly set in the configJEDA.cfg file (loaded at Jeda Start) the luminance in cd/m² is displayed when moving the Mouse over the Palette Drawing.

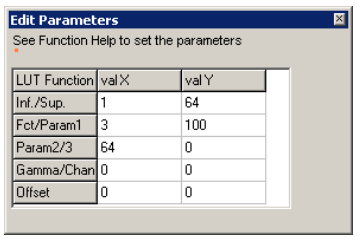
Jeda uses a function of the form:

$$L(cd/m^2) = offsetBlack + Slope*pow(bitRED, ExpoRED) + Slope*pow(bitGreen, ExpoGreen) + Slope*pow(bitBlue, ExpoBlue)$$

Calibration should fit the values measured in cd/m² (e.g. with a photometer) for each RGB gun to determine: the OffsetBlack (luminance for a black screen), the Slopes and Exponents for each gun, These parameters should then be set in the configJEDA.cfg file.

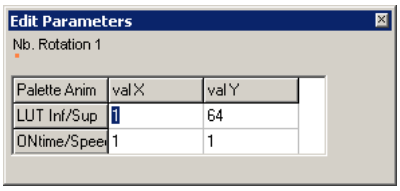
FUNCTION PALETTE

A function Palette is an Object setting the palette entries using a mathematical function. It is set each time it is called (in an Anim Object for instance).
The fields in the parameter Array are used to store the lower and upper limits –color indices- to be set by the function, the function to be used together with the 3 function parameters, the type of luminance distribution (Gamma or bits), the channel (Red/Green/Blue) and an Offset value. Last parameter is unused.



PALETTE ANIMATION

The Palette Rotation Object is used for animation of the colour table, but is only available in 8 bpp mode. Several palette rotation can be defined. Object Parameters are:
The colour limits for the palette rotation (Inf/Sup)
The On-time of each frame and the step to move in the palette (Speed).



TEXT

Text Object

Text Objects have several purposes:

1. A Text Object permits to encode and display a line of text on screen

Font Color, Size and Style (italic, bold etc.) can be modified using the dedicated fields.
FontName[13] "Monaco", "Times", "Arial", "MS Serif", "System", "Symbol", “Palatino", "Courier", "Script", "Garamond", "Wingdings 2", "Wingdings 3".
Styles :
1: Bold 2: Italic 3: Underline 4: StrikeOut 5: Bold and Italic 6: Bold and Italic and Underline
7: Text Invisible on Screen

2. When the text corresponds to a valid File Name of a RTF file, the file is loaded when the user exit from the Text Field and shown in a Message Box. The text can be modified (‘Edit’ Button) and is saved when the Message box is closed. One may Browse directories to find a FileName.

3. The field of a TEXT Object is also used to load Files, as the name of a Movie (AVI file) or of another Jeda *.eda file (Use the ‘Browse’ Button). In that case, the text is not displayed on Screen.

Test Box Object

A Test Box Object is a Device Contingent Object. It uses the XY coordinates of a device relative to the Box limits to perform operations depending on the result of the test.
A Box is defined by its:

- Color (0= invisible)
- Active (0/1)
- X Y size (Width/Height)
- Number of associate Objects (n)
- Its XY position on the screen

To help filling the fields of a Test Box Object a dedicated window opens to provide intuitive tools. Many Actions are possible: look at the C++ code in Wvisu.cpp / void TestBox().

The parameters of a Test Box are as follows:

Type : In or Out	Mode	Device (0-6)	Store if Reward given
Min Time in Box	Duration in Box	Store time in	Set Outcome
Action Reward (0-1-2;3,4)	Give	Pointer to next Box or Object	(ISI)
Associate Object #1	ID number	Action	---
… #2 etc.	-	-	---

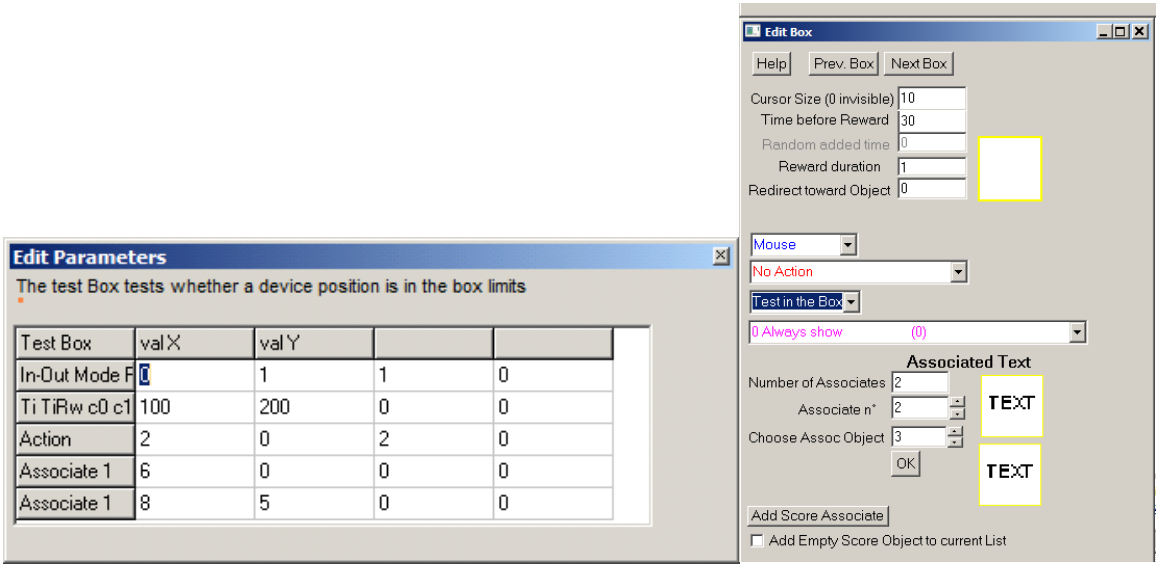
- Test Type: Test if IN or if OUT the box limits
- Mode: (0/1) display (or not) device position; 2 Associates attached to the Box position, 3 Idem without device feedback
- Device codes :
  Device Kind {none=0, keyboard=1, mouse=2, joystick=3, tablet=4, PortPLL=5, EyeLink=6, EyeTech=7, Eyecam=8, EyeTribe=9, TCPIP=10, DataIN=11, EyeSMI=12, EyeLT=13}
  Note: the chosen device should be enabled before editing.
  Note 2: the DataIN device uses a DotXYZT Object as input. The Dot Object may contains the coordinates of the recording made with another device (an EyeTracker) for replay and control.
- Min time in Box: accumulated number of frames before Test is positive
- Duration in the Box: number of frames during which the test remains positive and an Action should be performed
- Standard Actions: 0 return (test Result); 1: TTL Reward pulse; 2 Sound;
  - (3,4 shows Target position)

//----------

Associated Objects Action codes:
0: ON all time box active
1: Show if Test negative
2: Show if Test positive
3: Show if in the Box
4: Show if in out of the Box
5: Show if in the Box longer than minimum time
6: Show if in not the Box longer than minimum time
7: Show if not in the Box
8: Show if in the Box

Pupil controls (Z)

11: Give Reward if Pupil above threshold (and show target)
12: Give Reward if Pupil below threshold (and show target)
13: Give Reward if Pupil above threshold
14: Give Reward if Pupil below threshold
15: change Object color (or frequency for sounds)



Example of Parameters and Menu for editing a TestBox

Sound Mapper

A Sound Mapper is used to modulate a Sound (wave *.wav) using data from a Device. It is active when called by an Anim Object.

Device {none, keyboard, mouse, joystick, tablet, PortPLL, EyeLink, EyeTech, Eyecam, EyeTribe, TCPIP DataIN} CurrentDevice;

// code: 0 1 2 3 4 5 6 7 8 9 10 11

The following fixed parameters must be set (available in the Object Panel):

The #ID of a device (see above); The #ID of a target Sound; The size of a Window containing device data used to compute the speed/acceleration of the device (running average)

	P1	P2	P3	P4	
	x	**y**	**z**	**t**	**index**
Device #1	DD Param	Scale	Offset	DefValue	0
Sound	S-Feature	Low L1	High L2	Sign	1
Gaussian Dist	Sign (0, X)	Ampl	Sigma	OffSet	2
Device #2	DD Param	Scale	Offset	DefValue	3
Sound	S-Feature	Low L1	High L2	Sign	4
Gaussian Dist	Sign (0, X)	Ampl	Sigma	OffSet	5
Device #3	DD Param	Scale	Offset	DefValue	6
Sound	S-Feature	Low L1	High L2	Sign	7
Gaussian Dist	Sign (0, X)	Ampl	Sigma	OffSet	8
XY Reference	X extern	Y extern	Sound X	Sound Y	9
Devdata #1 Devdata #2 Devdata #3 …				DataSet1 DataSet2 DataSet3 - -	10

In the Object array (above):

The first 10 lines are used to set the parameters, and the remaining lines are used to store the device data.

The type of device data (DD Param) to be used for Sound modulation:

1: Device Position (XY) **1:X; 2:Y; 3: X,Y**

4: Distance of Device position to the reference positions passed by Anim function

5: Device Z (Pupil size if the device is an Eye Tracker)

6: Device speed (computed using N device data = Window Size-10)

7: Gaussian distance of Device to reference positions (Sign used for concave vs convex Gaussian : 0 or High value). Anim function can also be used to pass moving references.

The sound feature to be modulated:

#1: Intensity; Volume - **#2**: Pitch; Frequency - **#3**: Volume&Frequency **#4**: Pan; Left/Right

The mapping parameters, divided into:

Scaling factor (Scale, S), Offset, Default Value; Thresholds (Low/High limits, L1, L2),

Sign: indicates the sign of the modulation (sign=0 or the MaxValue)

The general formula to modulate a Sound is: Sound(#Feature) = Device(Param) x S + Offset

The formula is applied if: L1> Device(param) <L2, else the default Value is set.

Dataset indicate the computed value for each parameter

Reminder: With Sound Wave, Intensity=0 if Volume set to 10000

Serial Object

A Serial Object sets the parameters for a Serial communication to send text or data. A Serial Object must be used within an Animation or TestBox (specify the index of the Serial Object). In both cases, a device with a serial port (Arduino or Pure Data program) should be active and connections set.

The Serial communication is opened at Animation start with its parameters, and closed at Animation end.

Defined fields for a Serial Object (in the Object Panel):

1. Number of the Serial Port
2. Source of data to send:
   0: to use the values of an Anim function
   1: to use the values of theCurrent Device specified
3. Baud (Bytes/sec) from 1200 to 256000
4. Mode: 1 Out : 2: In (to read data)
5. A coding scheme used to parse the data, in the form:
   Ch1 X Ch2 Y Ch3 were Ch1 Ch2 and Ch3 are delimiters
   By default Ch1 =% Ch2= # Ch3=$
   Example of sent string
   %23#43$
   read by an Arduino program as X=23 and Y=43
   If no code is defined only the X value is sent.

A serial Object can be used in conjunction with a TestBox enabling a Device*.

The array panel shows the ascii code for parsing the data (line 1)

Line2 allows to set X and Y offsets, a scaling value and the size of a buffer for delayed actions.

The same coding scheme should also be used to decode the string to extract thee X and Y values, for instance with an Arduino (see code below).

Movies

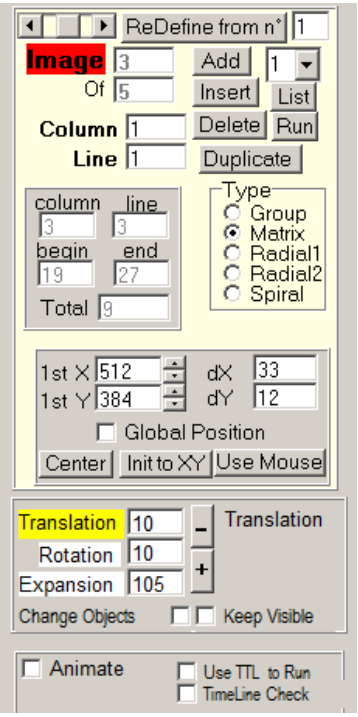
Jeda can display movies, depending on the Codecs that have been installed (load and install Codecs beforehand).
A movie can be loaded either in the Control Panel-> AVI Management sheet or in a Text Object (by entering the full pathname of the movie file).

The “Control -> AVI Management” panel provides information about the movie : type, size, number of frames, compressor etc.
If a decompressor is needed and is found the movie is shown; otherwise an error message is displayed.
It is possible to store a movie, frame by frame, into a 3DotXYZ object, by checking the appropriate “Make 3D cloud” checkbox. When checked, it becomes possible to indicate the sampling of the movie (1 default), the size of the images, and the Starting and Ending frames that are to be stored. Cropping is also possible. It is aso possible to choose a color that will be ignored (not stored).
Each frame of the movie is stored in a 3DotXYZ object that can be edited and modified (see utilities in the “Controls->3D Object Edit” panel). Uses a ListShow Object to show the frames in succession.

Alternatively, a Text Object can handle a movie, using its full name. In that case the TextObject handle a surface bitmap of the size of the movie, refreshed on each frame. Being a Jeda Object, the movie can be played using an Anim Object (and can be moved on screen). If the number of frames in the movie is smaller than the frames of the Anim Object, the movie is looped.

Making a Movie from an animation:
Jeda can make *.avi movies to export animation or sequences of images.
To do so: click on the “Record AVI Now” in the Control Panel; you can then launch an animation or change images on screen. Each image will be stored in the movie. Hit ESCape to end the recording.

Using Images



Images are usefull to set the positions of a collection of Object at once. Images define a first and a last Object to delimit the Objects within an Image. All Objects in between these limits belong to the current Image. Click on the I**mage Panel** to activate the Image Functions, or to check and to visualize edited Images.

Editing images proceeds in 3 steps:

1. Define the Image **Type** (**Radio Panel**) and its **Structure** (e.g. number of **Columns** and **Lines**),
2. Click on **Add** or **Insert**,
3. Define the Objects **positions**. Additional Objects are allocated as a function of the needs.

Different **Image Types** are available: Group, Matrix, Radial 1, Radial 2, Spiral (**Radio Panel**)

Group: a collection of several Objects

Matrix: Image defined by its number of columns and lines.

Radial1: Image defined by its number of Rings and Object per Ring

Radial2: Image defined by its number of Rings and its number of Object *for the first Ring*. The other Rings contain as many Objects as needed to fill each Ring (automatically generated by Jeda)

Spiral: Distribute the Image's Object along a Spiral.

Setting the Objects' positions :

1. Enter the X and Y coordinates of the **First Object** of an Image
2. Enter the Distance (**dX, dY**) between Objects on the X and Y axis

For **Radial Images**, set the position of the First Object, the **Angle** between Objects and the the **Distance** (Radius) between Rings

It is also possible to position the Objects of an Image with the Mouse (Button '**Use Mouse**'), by clicking on the Main Stimulus Screen. After a click an Object will occupy the Mouse Position, in succession.

Several Images can be positionned at once : Check the '**Global Position**' checkbox. The Objects' positions can be initialzed to the same XY position ('**Center**' and '**Init XY**' Buttons).

Changing Images

After ediding one or several Images, one can change their position, orientation and size using '**Translation**' , '**Rotation**', '**Expansion**' tools. 1. Choose the action by clicking on the label, 2, Set the **Increment** to use, 3. Click on '**+**' or '**-**' Buttons.

'**Translation**': click to select a translation on the **X** (yellow) or **Y** (red) axis

'**Rotation**': click to select a 'Rotation' of all objects (Yellow) or a distributed 'Rotation' (Red, in that case the rotation angle will be incremented for each Object)

If the '**Change Object**' checkBox is checked, the Objects will also Rotate or Expand.

An Image can also be moved by clicking and draging it on the **Small Control Screen**.

Duplicating Images

The 'Duplicate' Button duplicates the Image List: for instance if 2 images are defined,

clicking on 'Duplicate' will produce 4 images (similar to the initial 2 Images).

Important Note: Defining Images is necessary to perform an Experiment: Each Image will correspond to an Experimental Condition. Thus, one can first use the Edit Tools to set Objects' positions, **BUT** it might be necessary to redefine the images (**ReDefine from** n° #x Button) , to set up an Experiment, for instance when using an Animation or a Group Show as the *drivers* that show the Objects sequencing in a trial. In that case, the images should only include these Objects.

FILES SAVED by JEDA Editor

*.EDA FILES Jeda Object File
*.XYZ FILES List Of XYZT coordinates
*.TDT FILES List Of XYZ coordinates
*.BMP FILES Bitmap Files
*.AVI FILES Movie File

FILES to SAVE before JEDA EXPERIMENTS

SetUp *.rtf FILES defines the *.cfg, *.eda and *.exp files used for a RUN
*.EXP FILES File containing the parameters settings of an Experiment
*.CFG FILES File defining Directories, refresh rate see Jeda Settings
*.ESC Settings for a staircase
*.MRK TTL markers sent on Parallel Port (e.g. for MEEG)

FILES SAVED after a RUN of JEDA EXPERIMENTS

*.DAT FILES Summary
*.BRT FILES All trials with Key and Response times
*.AJU FILES All Trials with frame-by-frame information (see below)

Running an Experiment with Jeda

Main steps:

A. Building up stimuli in Jeda Editor.

Remember that each Object has an **Index** representing its position in the **List**. These indices are used as pointers to Objects.

Important: **During an Experiment an image corresponds to an experimental condition**.

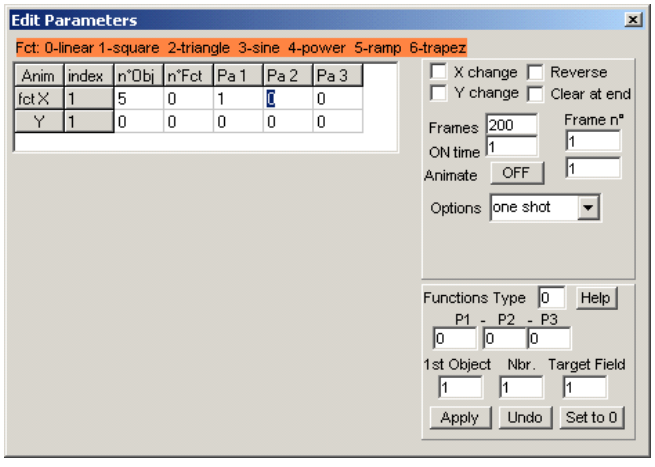
An image can be a single Object. Because Showers/Grabbers such as 'AnimObject', 'GroupShow' are used to display graphic objects in sequence, in motion or in groups, they are often used to make up an image with a single Object, i.e. an experimental condition.

Images/Conditions use the indices of Objects (their order or index) of an Object List:
Image1 uses Object #1 to #n, Image2 #n+1 to #m, Image3 from #m+1 to #k, etc.
For that reason, the Showers/Grabbers such as 'AnimObject', 'GroupShow' should be inserted in 1st, 2nd 3rd …etc. position in the list, so as to define Experimental Conditions

Example: Measuring the ability to detect Left-Right-Up-Down directions of motion.

1. Make 4 Anim Objects. (Indices 1;2;3;4)
2. Add a disk of chosen size and colour (index 5)
3. Fill the Parameter Fields of the Anim Objects

```
e.g. X    5    0   1   0   0
     Y    0    0   0   0   0
```



Anim 1moves object 5 to the Right at 1 pixel per frame for 200 frames

Define 4 Images/Conditions each having 1 Object. Each Image corresponds to a direction of motion and will be used as a single experimental condition during the Experiment. Save File: See `AnimExpeDir.eda` file

Objects useful to design an Experiments:

'Animation' permits to show Graphic Objects, Group of Objects, Random Dot Kinematograms, MoveAnim (a list of positions) and to move them with functions (Functions described in Help File). 3D Object are moved with a 3DAnim Object.

Group Show: A Group Show can display Objects in sequences of events with an ON time and OFF time (in number of frames). An 'Anim Object' can be displayed by a Group Show, using its own duration (number of frames).*

'Group': A Group shows one or several Graphic Objects defined by their lower and higher Indices. Several Groups can be shown at once. It is possible to show one or several Objects *chosen at Random* amongst the available Objects (enter the desired value of Object to show simultaneously in the "All Objects" field of the Group.

See Group Object above and example: `Jeda/Images/DemoJeda/RandomGroup.eda`

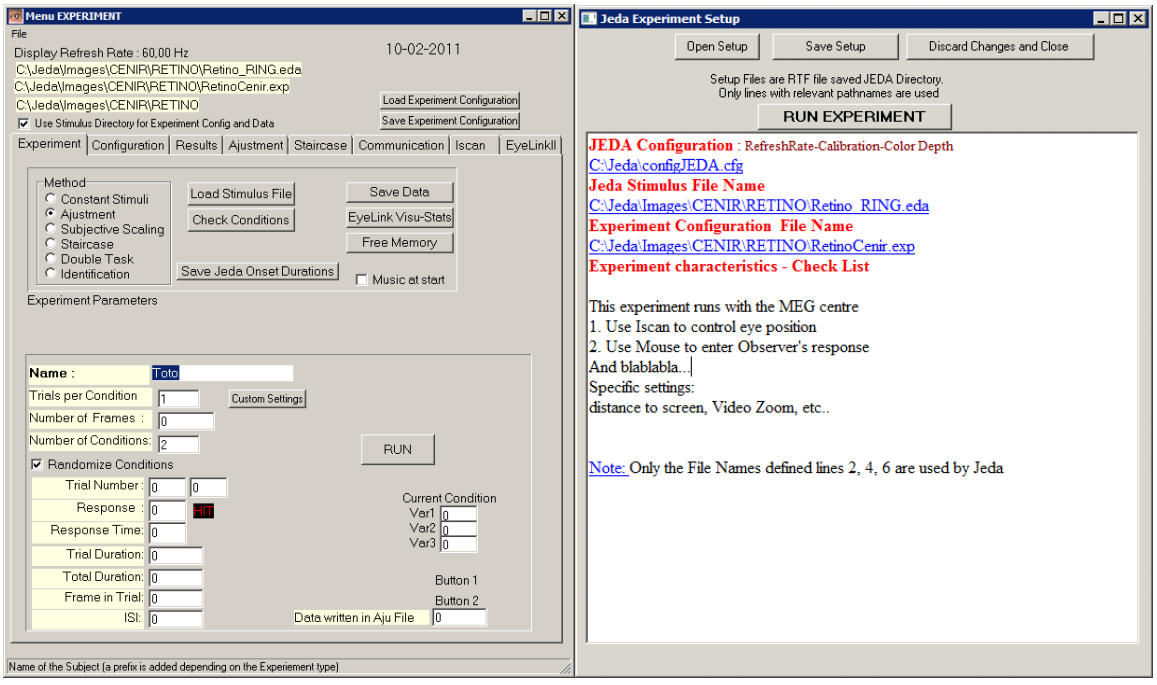
B. Running An Experiment

1. Load an EDA file for the experiment (e.g. `AnimExpeDir.eda`)
2. Select the Experiment Menu in the Main Menu:

The 1st time

1. Set up the Experimental Configuration
2. After all settings are chosen, save the Config.exp File with a New Name (e.g. ExpeAnimDir.exp)
3. Optionally you can edit a *.rtf File and set the names and directories as shown in the example below

---

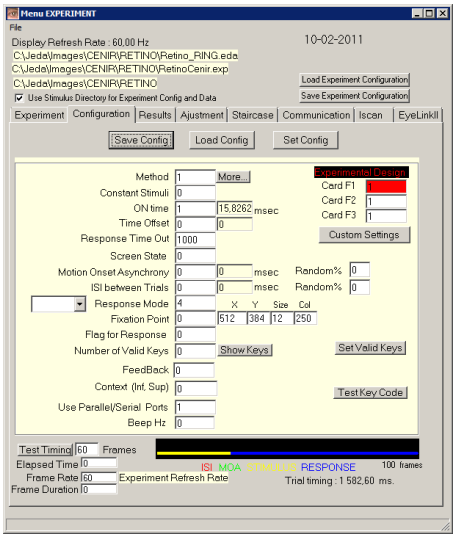


Next times: Load the *.rtf File from the Run Setup in the Main Menu and Click Run EXPERIMENT. Enter Subject Name and click RUN.

Setting up the Experimental configuration:

1. Select the method: Constant Stimuli, Adjustment, Staircase, etc..
2. Go to "Configuration" and choose the parameters for the experiment:

a. ON time (1 by default) in frames (duration of each frame)
b. ISI in frames and RandomISI
c. Fixation point (On/Off, Type, size, color..
d. Response Mode: KeyPress, NoResponse, etc..
e. … see help bubbles



*After all values have been set, Save the *.exp File.*

The Adjustment and Staircase Methods require additional settings:

Staircase

For the staircase method a special (*.esc) file must be saved –and reloaded at Run Time, after settings of the staircases have been made.

Choose the number of simultaneously running staircases (chosen at Random at Run Time). For each staircase define:

1. Convergence Value, Starting Value, Step, Number of Inversions to average, Number of variables runs (division by 2 of the Step for the N variable runs)
2. A Target Object -the Object or Object field to be modified by the Staircase- using its Index and a Target Field: the field to be modified: radius, colour, etc.. Extra field values (e.g. Contrast of a Gabor Patch) are accessed through a Column/Line indexing: column X or Y, then the index in the Field Array (starting with 0).
3. Define the Correct Key: hitting this key will be considered a Hit, and used to Up-Down the selected Target Value.
4. Save the Staircase settings File

N.B. If the 'RecordData' is checked, responses will be continuously recorded during a trial and saved at the end of a RUN

Adjustment Method

The Adjustment Method permits to record all features of each frame during an experiment. It is also used for adjustment purposes (e.g. adjust the size of a Disk, point with an arrow, etc..) A file with extension *.aju is written on Disk.

File *.aju : These files contain a frame by frame follow-up of the successive trials of an Experiment. These files are arrays of 24 columns, each storing different relevant fields

Header: Number of Lines in file, Number of Trials (2 integers)
Trial: Trial Rank, N° Condition, Reserved, Lines in Trial (5 integers)

```
%         1       2       3              4         5         6
% DeviceX  DeviceY  DeviceZ  Button1  Button2  Button3
%       7             8             9
% StimPosX  StimPosY  StimPosZ
```

```
%    10     11      12
% Red   Green   Blue
%        13         14              15            16      17       18
% StimIndex1  StimIndex2  StimIndex3    Menu    Frame    Time
%    19         20         21         22         23         24
% EyeH1    EyeV1    Pup1     EyeH2    EyeV2    Pup2
...
```

A MatLab script is available to convert these Aju Files to Mat Files for further analyses

DATA & FILES saved at the End of a RUN

2 TEXT Files are saved at the end of a RUN. These files can be loaded with Excel for calculations. Their names are made of a prefix: C_ for constant Method, A_ for adjustment, S_ for Subjective, E _ for StairCase, D_ for double Tasks, I_ for identification, and the Subject Name.

File *.dat contains the Name Date, StimFileName *.eda, ConfigFileName *.exp.
With the Method of constant stimuli, the percent correct and response time for each condition is provided

```
Name..juan_reco          13-5-2005        Ficher eda  C:\Jeda\Images\Sofia\recognition.eda
Repetitions x FA1 x FA2 x FA3=N Trials
   1 x 160  x  1  x  1= 160
Config File: C:\Jeda\Images\Sofia\recognition2 Exp Sofia.exp    MOA:  0   ISI:150          ONtime:  1
Mode Rep:0       Pt Fix:0
Trial  Duration: ..133.188 s --- Refresh Rate 150.176
Cond.   %Correct   RTc     %Error    RTe
 1  1         0.00      0.00       100.00    806.00
 2  2         100.00  2825.00     0.00     0.00
 3  3         100.00  1515.00     0.00     0.00
 4  4         100.00  3878.00     0.00     0.00
 5  5         0.00       0.00      100.00   1066.00
 6  6         100.00  1227.00     0.00     0.00
 7  7         0.00      0.00       100.00   625.00
 8  8         100.00  1358.00     0.00     0.00
 9  9         0.00      0.00         100.00   737.00
```

File *.brt contains all data of each trial. It can be used to calculate statistics knowing the Key Code and the Experimental Condition (CdF1 in the example below)

```
Nom..zoe_entr        26-5-2005          Ficher *.eda
C:\Jeda\Images\Sofia\HAND.eda
Repetitions x FA1 x FA2 x FA3=N Trials
    1 x 12  x  1  x  1=  12
Config File : C:\Jeda\Images\Sofia\Visage Exp Sofia.exp    MOA:  0       ISI:150
       ONtime:  1
Mode Rep:0    Pt Fix:0
Trial Duration..6.659 s --- Refresh Rate 150.182
Resp      RT     CdF1       CdF2       CdF3    Rank   Resp. Ascii Code
  0      2338      1          0          0        5        49
  1      2419      2          0          0       12        51
  1      2488      3          0          0        4        51
  1      2458      4          0          0       11        51
  1      2506      5          0          0        6        51
  0      2483      6          0          0        8        49
  0      2530      7          0          0        1        49
  0      2753      8          0          0        9        49
  0      2344      9          0          0       10        49
  0      2328     10          0          0        3        49
  0      2372     11          0          0        2        49
  0      2434     12          0          0        7        49
```

Communication with external devices

Communication with other devices:

It is possible to send a Pulse on the Parallel port (Jeda is then the Master of the Experiment: e.g. MEG/EEG) or to wait for a Pulse (Jeda is then a slave during the experiment –e.g. fMRI, CENIR)
It is also possible to read or send data on the serial Port.

Eye movements:

- EyeLink (connect the EyeLink PC and the Jeda PC through Ethernet).
- EyeTribe (USB 3.0)
- LiveTrack (USB, CRS Research)
- SMI Red (USB 2.0)

A TTL Object is also available in the scrolling Object list. Each time it is called it sends a TTL pulse with a specific value on the parallel port
A Serial Object can be edited to enable Serial communication (e.g. with Arduino board)

CONFIG Settings

The ConfigJEDA.cfg file contains information relative to Jeda Directories, Dual/Single screen mode, Pixel size refresh rate, Bit-Depth (8 bpp 256 or 32 bpp 16 million colors, PIXEL=8 or PIXEL=32), and Gamma correction/Screen calibration settings

A typical configuration file is like this (without the comments):

```
-------------------------------------------------------------------------
c:/Jeda/Images/
c:/Jeda/ConfigExp/
c:/Jeda/Results/
c:/Jeda/Film/
c:/Jeda/Bitmap/
c:/Jeda/JedaBMP/
c:/Jeda/JedWav/
75      // put 1 for Single Screen mode; 2 or the refresh rate (e.g. 75 85, 120 Hz) for double screen mode (using
        extended displays. See Special MODES below)
 0.035              // pixel size in cm
                    // values for gamma correction :
                    //Function used for Gamma each gun: L=offsetBlack+ Slope * pow(BIT, exponent)
0.00130000          // Offset Black (luminance in cd/m² when R=0 G=0 B=0
0.00001537          // Red Exponent
2.52330000          // Red Slope
0.00003310          // Green Slope
2.60175000          // Green Exponent
0.00000507          // Blue Exponent
2.61785000          // Blue Slope
EOL0                // You may use a Flag to modify Jeda's appearance at start : EOL1 is used to Play EyeGames
                    // PROTOC1 launches the Protocol interface of Jeda
-----------------------------------------
// Special MODES
    GUID 1: MAXX=1024; MAXY=768  PIXEL=32    SingleScreen=1
    GUID 2: MAXX=1024  MAXY=768    DualScreen=2    RefreshRate=75 Hz
    GUID 3: MAXX=640    MAXY=480    DualScreen =2
    GUID 4: MAXX=1280  MAXY=1024  DualScreen =2
    GUID 5: MAXX=1600  MAXY=1200  DualScreen =2
    GUID 6: MAXX=1024  MAXY=768   DualScreen =2   RefreshRate =150 Hz
    GUID 7: MAXX=1024  MAXY=768   DualScreen =2   RefreshRate =85 Hz
    GUID 8: MAXX=1024  MAXY=768   DualScreen =2   RefreshRate =100 Hz
    GUID 9: MAXX=1024  MAXY=768   DualScreen =2   RefreshRate =60 Hz
   default: MAXX=1024  MAXY=768   RefreshRate=GUID value; DualScreen =2    PIXEL=32;
```

FIELDS in Objects ARRAY

Jeda uses a single structure to store the specific fields of different objects. Knowing the correspondence between the Objects'Fields and the arrays can be useful to determine their address/coordinates (e.g. to change a parameter of an object list at once using a function)

Correspondence between parameters and Object array in JEDA:

Parameters of objects are stored into Arrays (TAB (x, y, z, t). The relationships between Parameters and Array indices are described below.

Random Dots Kinematograms:

```
  Number of frames: angle;
  ONtime      :       length;
  Area Type   : tab[0].x;
  SizeX       : tab[1].x;
  SizeY       : tab[2].x;
  PosX        : tab[3].x;
  PosY        : tab[4].x;
  Mask        : tab[5].x;      // Mask Object
  Dot Shape   : tab[0].y
  DotSizeX    : tab[1].y;
  DotSizeY    : tab[2].y;
  DotCol      : tab[5].y;
  DotNumber   : tab[3].y;
  Object      : tab[4].y;      // Target Object
  Functions X: tab[6].x;
     P1x      : tab[7].x;
     P2x      : tab[8].x;
     P3x      : tab[9].x;
  Functions Y: tab[6].y;
    P1y       : tab[7].y;
    P2y       : tab[8].y;
    P3y       : tab[9].y;
% Coherence   : tab[10].x;
RDK Mode      : tab[11].x;
Direction     : tab[12].x;
Speed         : tab[13].x;
RangeDir      : tab[14].x;
RangeSpeed    : tab[15].x;
LifeTime      : tab[10].y;
Replot Mode   : tab[11].y;
ZObject       : actu->tab[13].y;
Z transparent: tab[14].y
Reserved : actu->tab[12].y=0;
```

Object Animation:

```
 Number of frames        : angle;
 ONtime                  : length;
 Morphing Anim :
        Number of Animation to morph     : mod_affich
        Number of steps                  : couleur
```

X	tab[0].x	tab[1].x	tab[2].x	tab[3].x	tab[4].x
tabX	Object index	N° Function	P1	P2	P3
Y	tab[0].y	tab[1].y	tab[2].y	tab[3].y	tab[4].y
tabY	Special code	Function	P1	P2	P3

Correspondence between Objects and Code numbers:

```
#define JDOT 1
#define JLINE 2
#define JCIRCLE 3
#define JDISK 4
#define JeRECTANGLE 5
#define JfRECTANGLE 6
#define JerPOLYGONE 7
```

```
#define JfrPOLYGONE 8
#define JPOLYLINE 9
#define JeiPOLYGONE 10
#define JfiPOLYGONE 11
#define JeELLIPSE 12
#define JfELLIPSE 13
#define JTEXTE 14
// pointeur d'Objets
#define GROUP 15
//----------
#define JGABOR2D 16
#define JGABOR  17
#define JPLAID 18
#define JGRATING 19
#define JSPIRALES 20
#define JBITMAP 30
#define JDOTXYZ 40
// pointeurs d'Objet
#define JMATRIX 41
#define JMASK 42
#define JMASKXY 43
// ------------
#define JFRACTAL 50
#define JSOUNDWAV 100
#define JSOUNDpause 101
#define JROTPALLETTE 131
// pointeur  d'Objet
#define JFANIM 132
#define JFMOVE 133
#define JFANIM3D 135
#define JRANDOMDOTS 150
#define JSHOWGROUP 222
#define JSHOWLIST  223
#define JSHOWIMAGE 224
// ---------------
// COLOR PALETTE
#define JFUNCTIONPALETTE 254
#define JPALLETTE 255
//------
#define JGRAPHICSGLOVE 256
#define JTTL 257
```
