



# Workshop4 PRO

## SMART WIDGETS EDITOR

# USER GUIDE

**Document Revision: 1.1**  
**Document Date: 29<sup>th</sup> July 2017**

## Table of Contents

---

<b>1. Description .....</b>	<b>4</b>
<b>2. Requirements .....</b>	<b>4</b>
<b>3. Opening the Tool.....</b>	<b>5</b>
<b>4. Starting a New Widget .....</b>	<b>6</b>
4.1. Size of the Working Area .....	8
4.2. Location of the Base Image in the Working Area.....	11
<b>5. Adding Layers.....</b>	<b>13</b>
5.1. Alpha.....	14
<b>6. Organizing Layers .....</b>	<b>15</b>
6.1. Order of Layers .....	15
6.2. Swapping Contents of Layers .....	16
<b>7. Setting Layer Parameters .....</b>	<b>20</b>
7.2. Layers with a Manipulated Image .....	20
7.2.1. Horizontal Motion.....	20
7.2.2. Vertical Motion .....	25
7.2.3. Diagonal Motion .....	28
7.2.4. Angular Motion .....	31
7.2.5. Combination of Diagonal and Angular Motions .....	42
7.3. Layers with Multiple Fixed Images .....	45
7.3.1. Frames and Values.....	47
7.3.2. Display the Static Images .....	48
7.3.3. Change the Location of the Static Image .....	49
7.4. Layers with a Series of Manipulated Images.....	50
7.5. Adding Numerical Part .....	54
7.5.1. Display Options .....	55
7.5.2. Changing the Font Colour.....	56
7.5.3. Changing the Font, Font Size and Font Style .....	57
<b>8. Linking Layers .....</b>	<b>58</b>
8.1. Clockwork Link.....	62

<b>9. Display Widget Frames</b> .....	<b>65</b>
9.1. FAT16 File Size Limitation.....	65
9.2. Scaling Widgets .....	66
9.2.1. Disable Proportional Scaling .....	67
9.2.2. Resampling Options.....	68
<b>10. Exploring Possibilities</b> .....	<b>68</b>
10.1. Workshop4 Examples .....	69
10.2. Application Notes .....	70
<b>11. Revision History</b> .....	<b>71</b>
<b>12. Legal Notice</b> .....	<b>72</b>
<b>13. Contact Information</b> .....	<b>72</b>

---

## 1. Description

This document describes the Smart Widgets Editor tool.

The Smart Widgets Editor tool is a powerful utility that can be used to create complex widgets with up to six layers, including the layer for the base image. The layers can be arranged in any desired order. Each layer can contain one or more images arranged in sequence that can be manipulated in a variety of ways.

Image manipulation options include horizontal motion, vertical motion, and angular motion. Two or more layers can be linked to produce synchronized movements of images. This enables the generation of complex widgets with multiple moving parts.

## 2. Requirements

The following are required before using the Smart Widgets Editor tool:

- Windows PC (Windows XP to Windows 10)
  - Windows 7+ is ideal
- Workshop4 IDE installed

**Note:** Before using Smart Widgets Editor, it is required to purchase a Workshop4 PRO license. This will then enable the Smart Widgets in the widgets pane.

### 3. Opening the Tool

In the widgets section of Workshop4 Pro, placing Smart Gauge from the Gauges Tab,



Smart Gauge

a Smart Slider or a Smart Knob from the inputs tab



Smart Slider



Smart Knob

will create the an empty object named accordingly with a property named "Config".

Property	Value
Name	SmartGauge1
Alias	SmartGauge1
Config	SmartGauge1
Left	00
Top	20

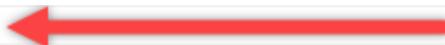
Clicking this button to open the tool

SmartGauge1



Double-clicking on the entry will also open the tool.

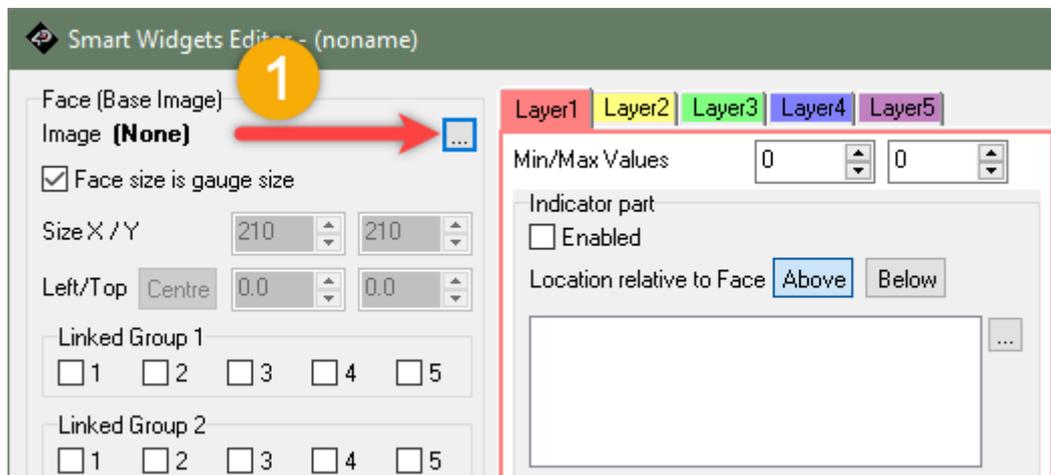
Config	SmartGauge1
--------	-------------



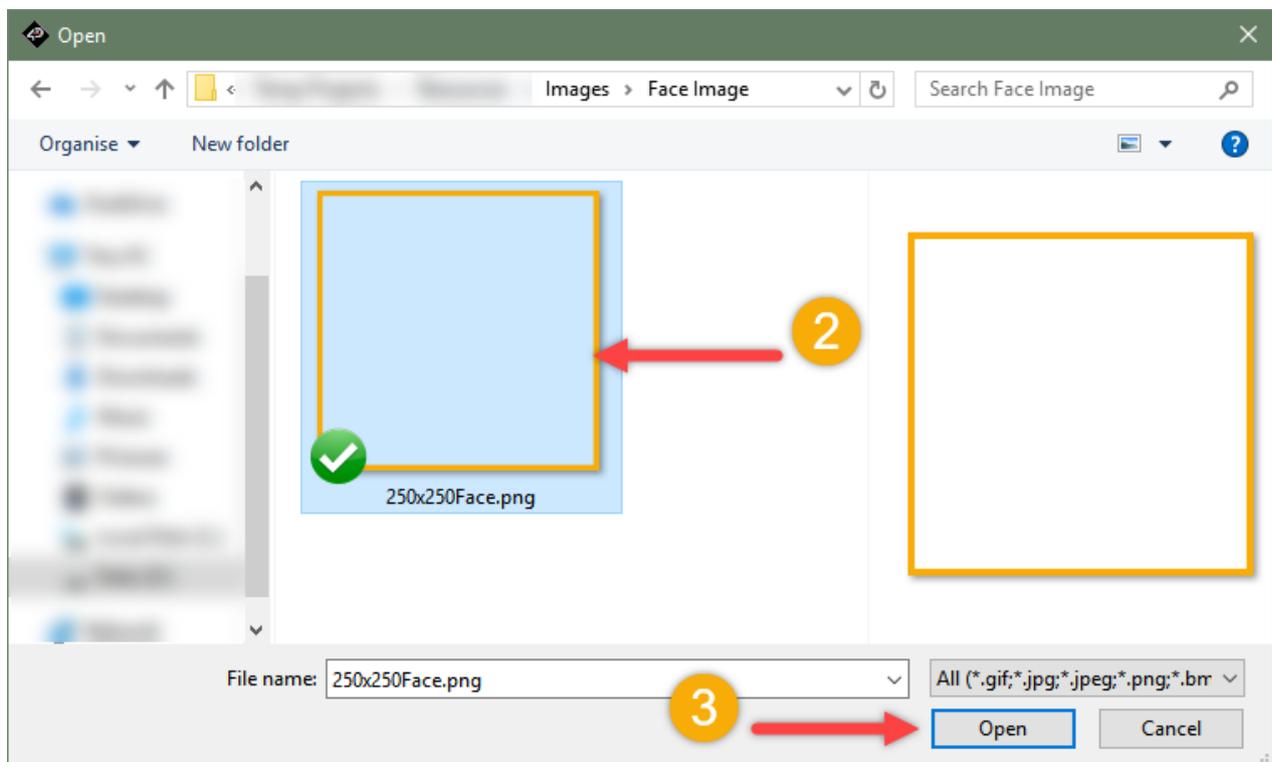
If the project hasn't been saved at least once, opening the tool will require you to do so before continuing.

## 4. Starting a New Widget

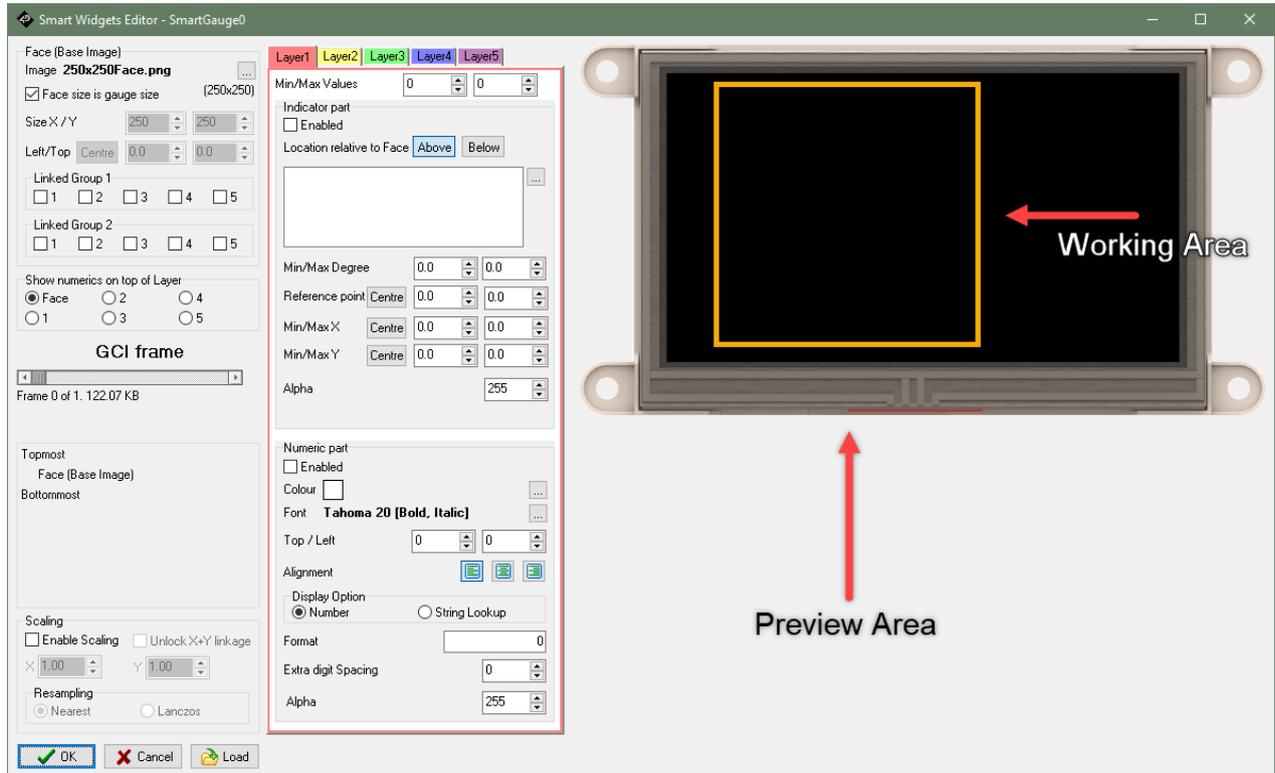
To start creating a new widget, add a face or base image. Click on the ellipsis symbol indicated in the image below. The face or base image is a static part of the widget.



Select the desired file. In this example, a 250x250 pixel PNG image with a blue border and a transparent interior is chosen as the base image.



The face image now appears on the preview area.

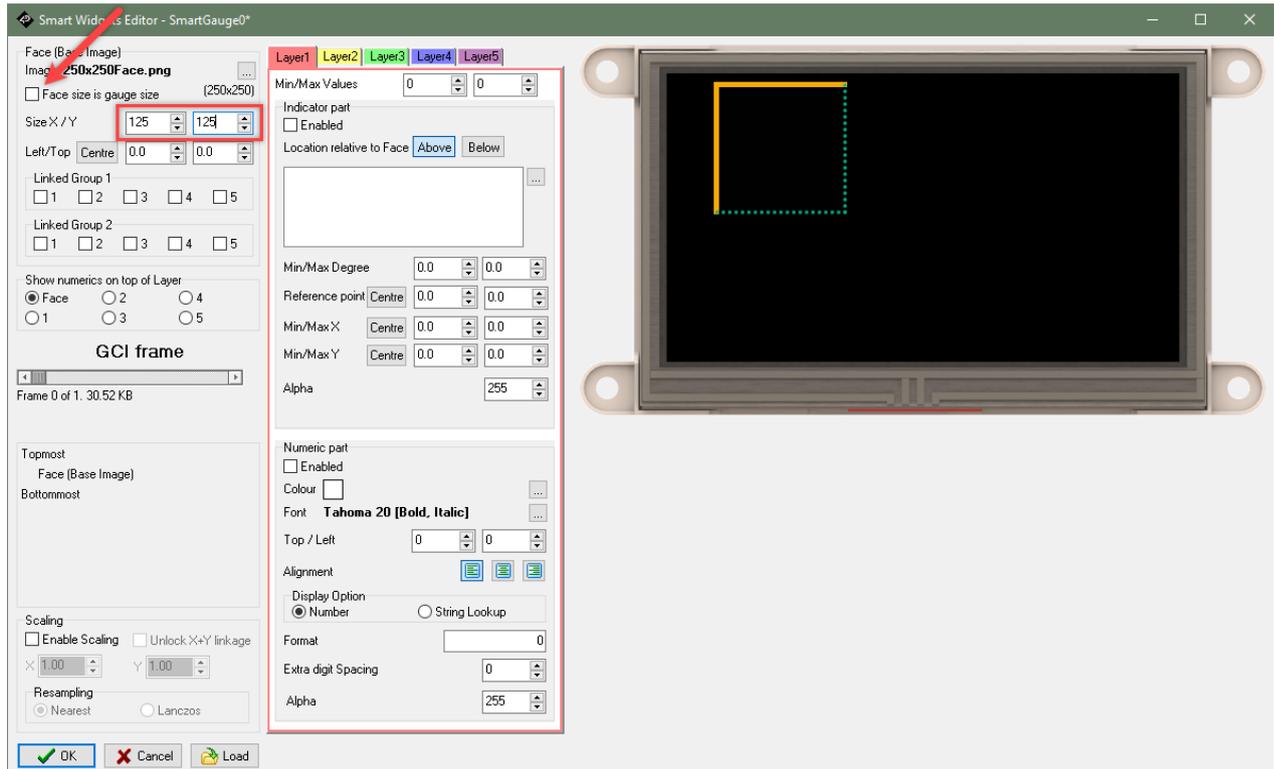


The **Preview Area** is the size of the whole 4D Display. The **Working Area** covers the whole widget area that is the size of the Face Image by default.

## 4.1. Size of the Working Area

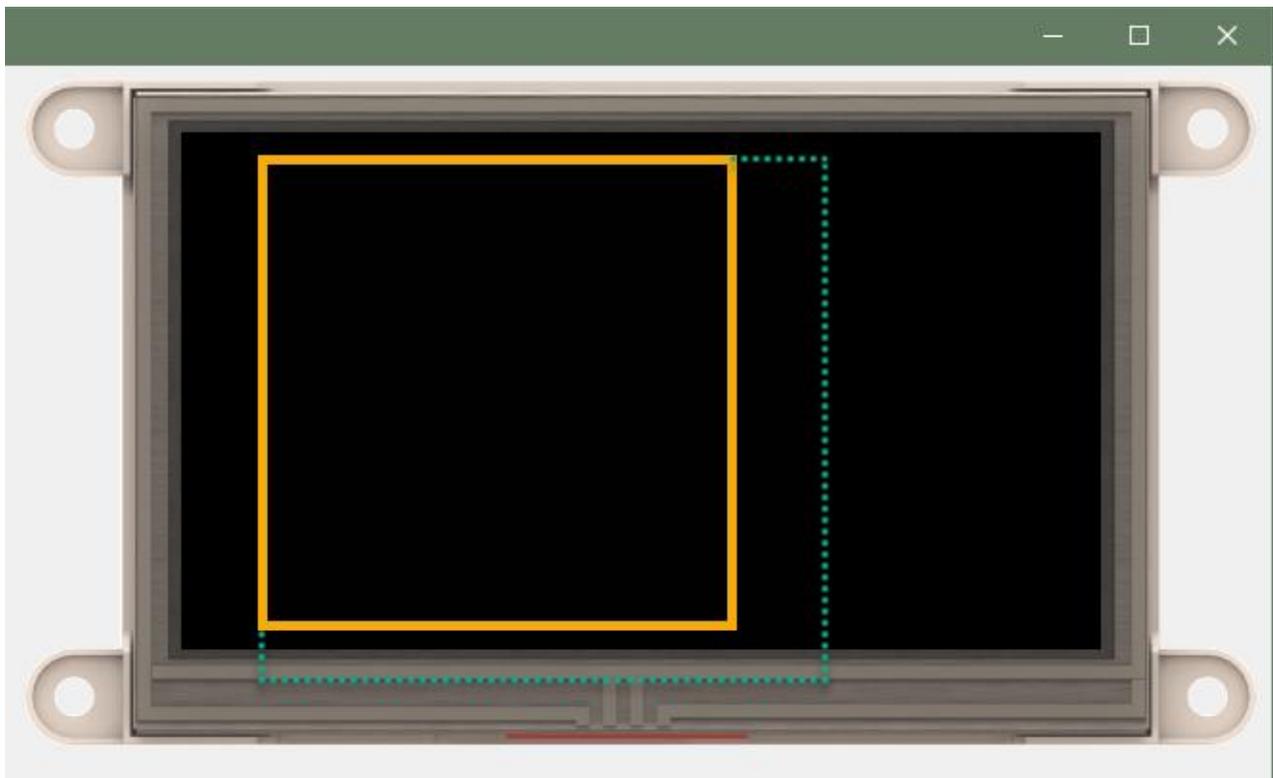
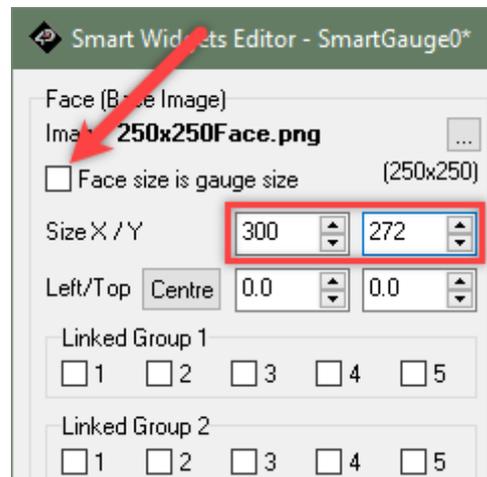
By default, the Smart Widgets Editor tool assumes that the working area is the entire size of the newly added base image.

It is possible to limit the size of the working area to a smaller section of the base image. To do this, remove the tick mark from the **Face size is gauge size** option and set the values of the parameters **Size X/Y** to a value less than the dimensions of the base image. In the example below, the working area is half of the whole base image.

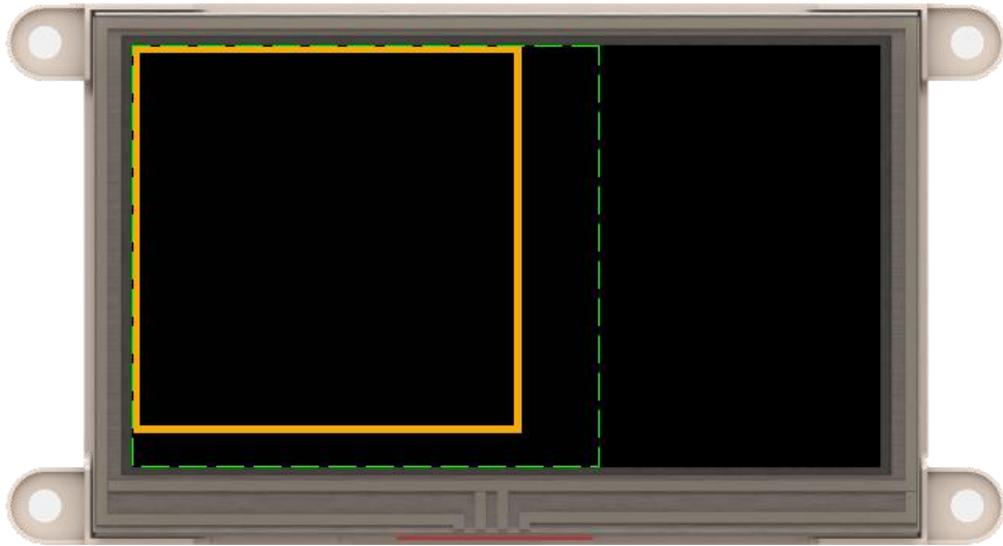


**Note:** The dotted lines in the above image serve to illustrate the resized working area. These do not actually appear on the preview area.

It is also possible for the working area to be larger than the size of the base image. To illustrate, the parameters **Size X/Y** are set to **300** and **272** respectively.



**Note:** The dotted lines in the above image serve to illustrate the resized working area. These do not actually appear on the preview area.

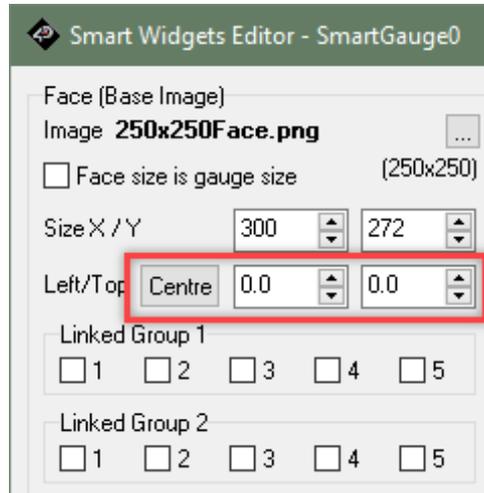


It can be observed better in the WYSIWYG area.

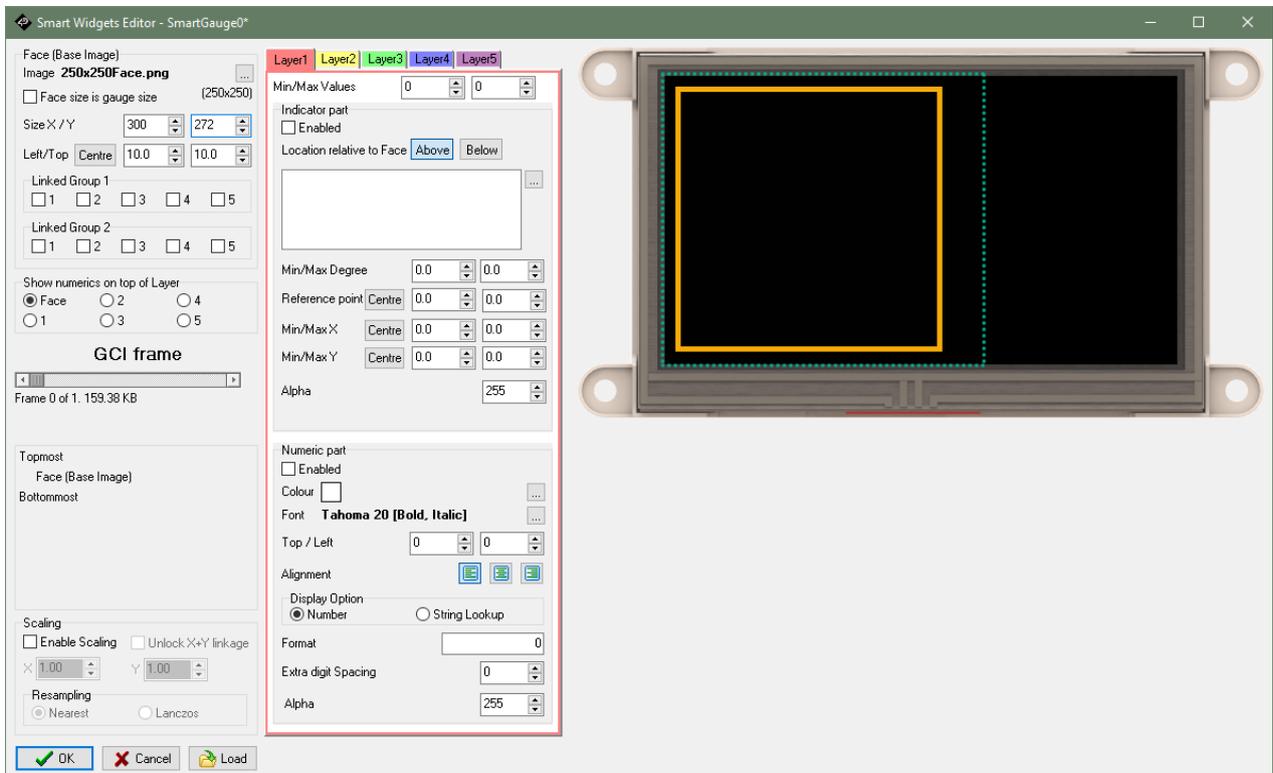
Noticed that in the above image the widget's position is also changed with its top left at (0, 0).

**4.2. Location of the Base Image in the Working Area**

Following the example from the previous section, if the size of the working area is set manually, the face/base image can also be repositioned.

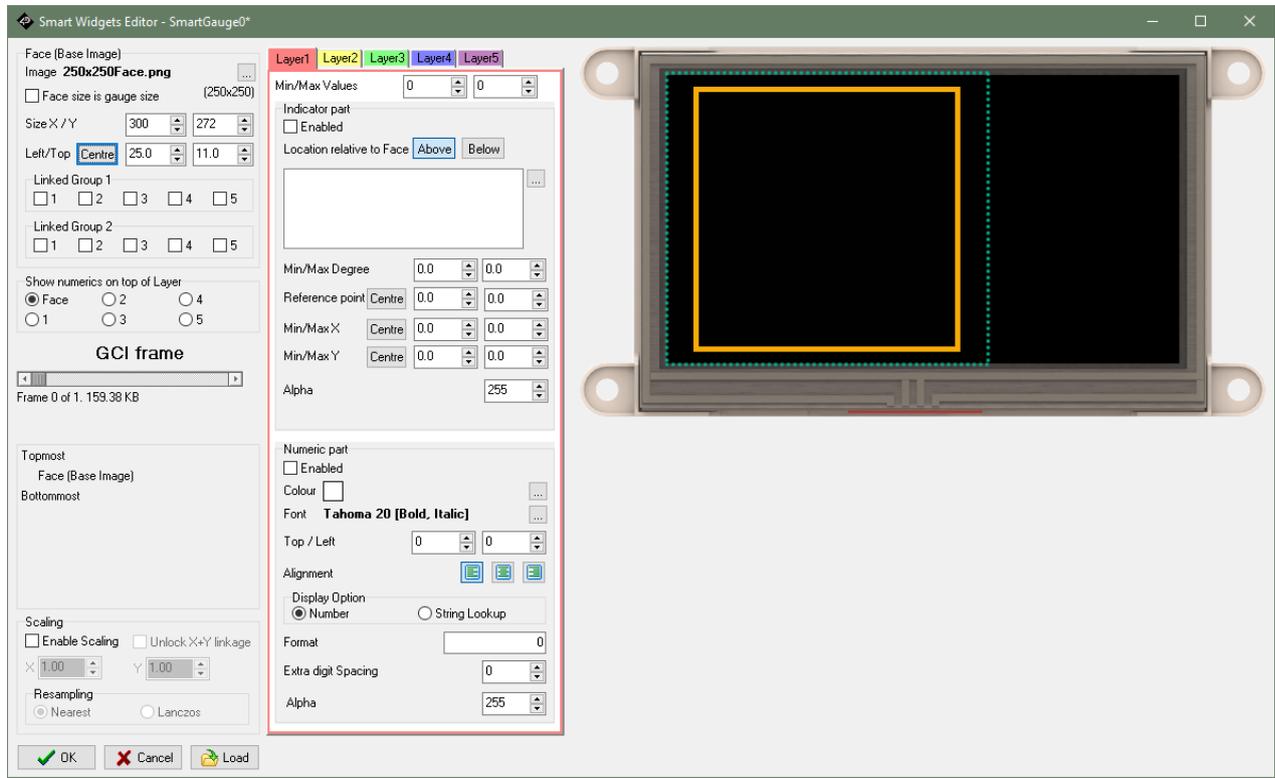


To change the location of the base image in the working area, set the values of the parameters **Left/Top**. These are the coordinates of the top-left corner of the base image with respect to the origin of the working area. As an example, the **Left/Top** parameters are set to **10** below. The base image is therefore moved to the right by **10** pixels and downwards by **10** pixels.

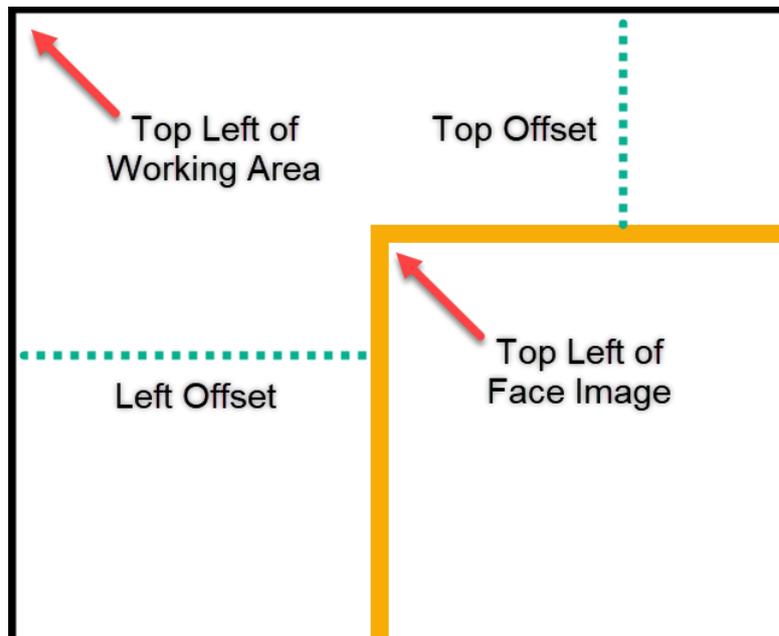


**Note:** The dotted lines in the above image serve to illustrate the resized working area. These do not actually appear on the preview area.

To move the base image to the centre of the working area, click on the **Centre** button as shown below. Smart Widgets Editor automatically computes for the **Size X/Y** parameter values.



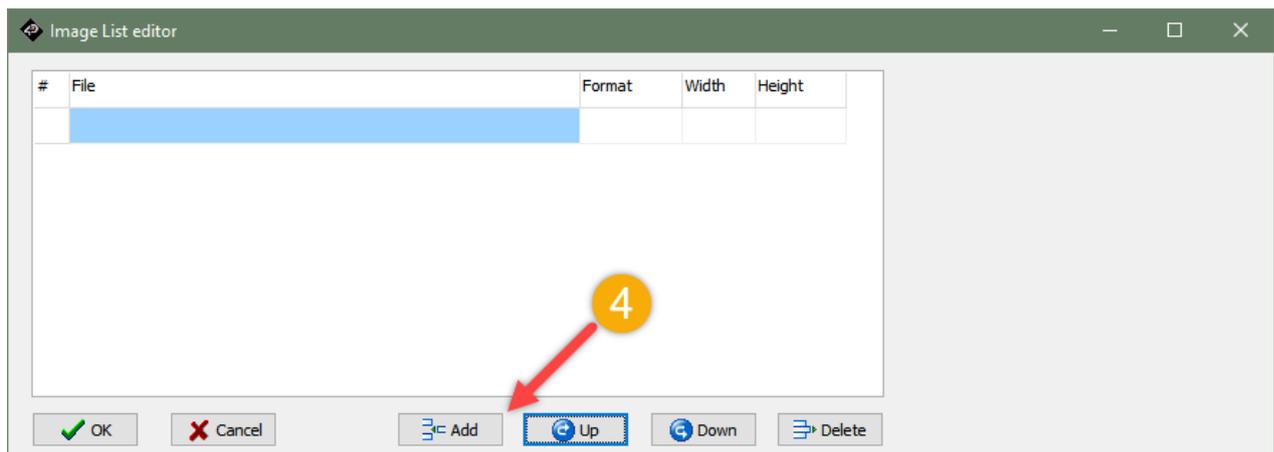
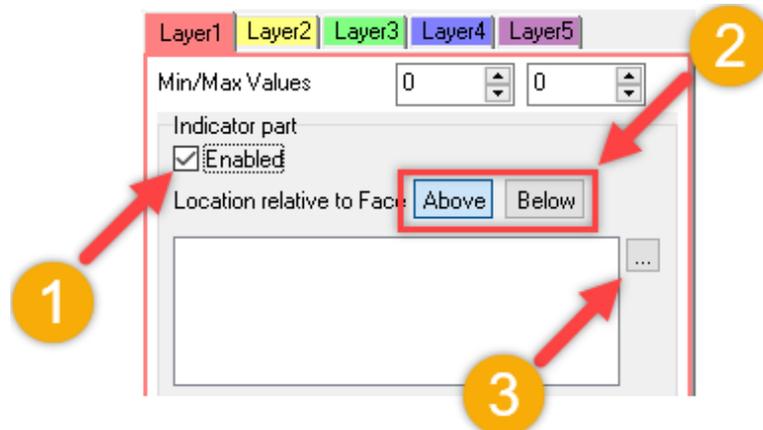
**Note:** The dotted lines in the above image serve to illustrate the resized working area. These do not actually appear on the preview area.



## 5. Adding Layers

The face or base image is a static part of the widget to which another separate part/layer, image or group of images, can be added. To add another image or group of images, a layer must first be created.

To add a layer, follow the steps indicated below.

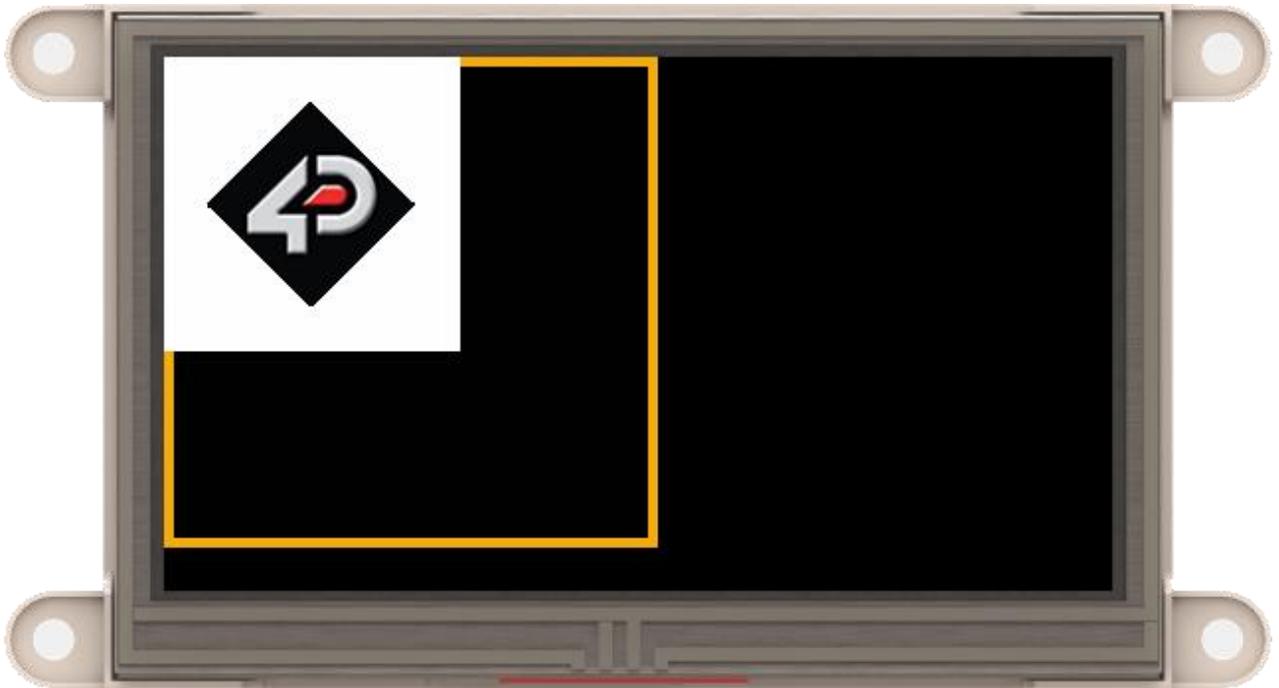


Layers can contain single or multiple images that can be manipulated in a variety of ways.

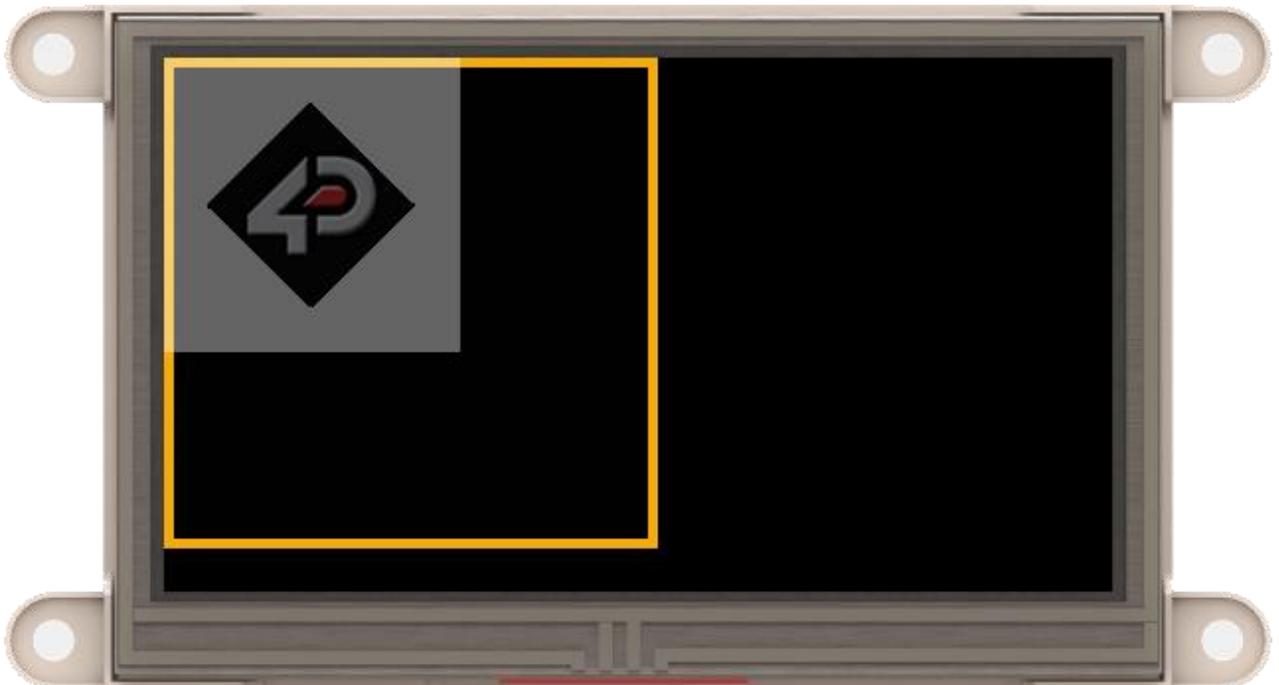
It is possible to manipulate a single image's horizontal position, vertical position and angular rotation, show multiple fixed images and manipulate multiple images that change in sequence. All of these are discussed in detail in the sections [Layers with a Manipulated Image](#), [Layers with Multiple Fixed Images](#) and [Layers with a Series of Manipulated Images](#) respectively.

## 5.1. Alpha

The transparency of a layer can be adjusted by varying the value of its *alpha* field. By default, a layer is opaque and has an alpha value of **255**. Below is a layer with the alpha value set to **255**.



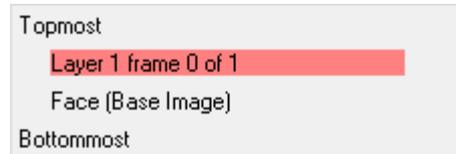
Below is the same layer with the alpha value set to **100**.



An alpha value of **0** means that a layer is fully transparent.

## 6. Organizing Layers

Smart Widgets Editor allows the user to use up to five layers of images, besides the layer for the base or face image. When working with multiple layers, it is best to refer to the layer guide to determine the order of arrangement of the layers, from top to bottom. The layer guide is indicated below.



In the above example, there is only one active layer (Layer 1) besides the layer containing the face/base image. Also, note that Layer 1 here is above the layer containing the face/base image. Each layer can be configured to be on top or below the layer containing the face/base image.

### 6.1. Order of Layers

For all layers below the base image, the layer with the lowest index number is always the topmost layer while the layer with the highest index number is the bottommost layer. The same applies for layers above the base image.

Note that Smart Widgets Editor automatically generates the frames of an enabled layer, based on how the layer is configured. For a widget containing a single enabled layer, the frames of the final output widget will be the generated frames of the single layer. For a widget with more than one enabled layer, each frame of the final output widget is actually a combination of the corresponding frames of each enabled layer.

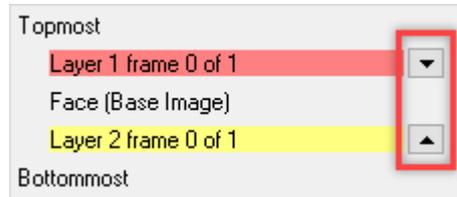
It is important to note that Smart Widgets Editor generates all possible combinations of each enabled layer as the output frames of the final widget. The frames are generated in a cyclic and sequential manner such that the layer with the lowest index number completes a cycle with all of its frames before using the next frame of the layer with the second lowest index number. This process continues such that when the layer with second lowest index number completes a cycle with all of its frames, the next frame of the layer with the third lowest index number is used.

The process described above can be easily visualized with a clock. The second hand having the lowest index number, the minute hand having the second lowest index number and the hour hand having the highest index number. The second hand must first complete a cycle before the minute hand move by one minute and the minute hand must first complete a whole hour for the hour hand to completely move by one hour.

It is also possible to link two or more enabled layers. Linked layers can be viewed as a single layer with index number equal to the index number of the layer with the lowest index number in the group. For more details, refer to the section [Linking Layers](#).

## 6.2. Swapping Contents of Layers

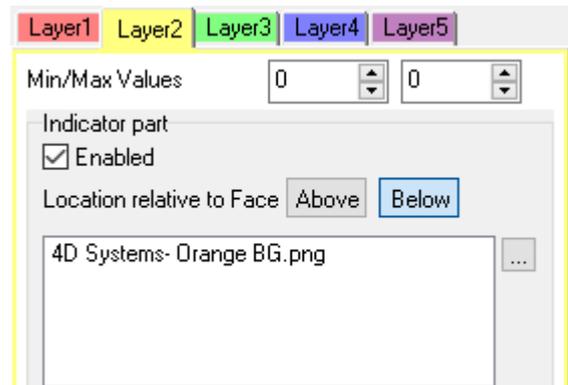
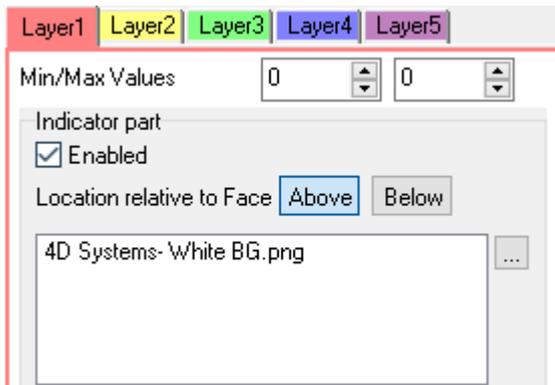
The contents of two adjacent layers can be swapped by using the up/down buttons in the layer guide. This doesn't include the face/base image. Using this technique, the contents of all layers can be rearranged as desired.



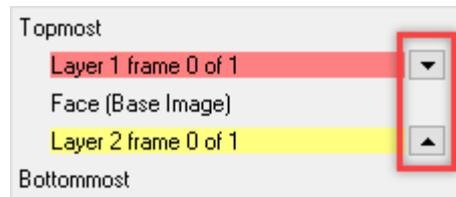
To facilitate this discussion, another layer is added to the project. Layer 2 contains an image again with 4D Systems logo but this time with an orange background. This layer is configured to be below the face/base image layer. The **Top/Left** parameters of this layer are also configured accordingly to come up with the output below.



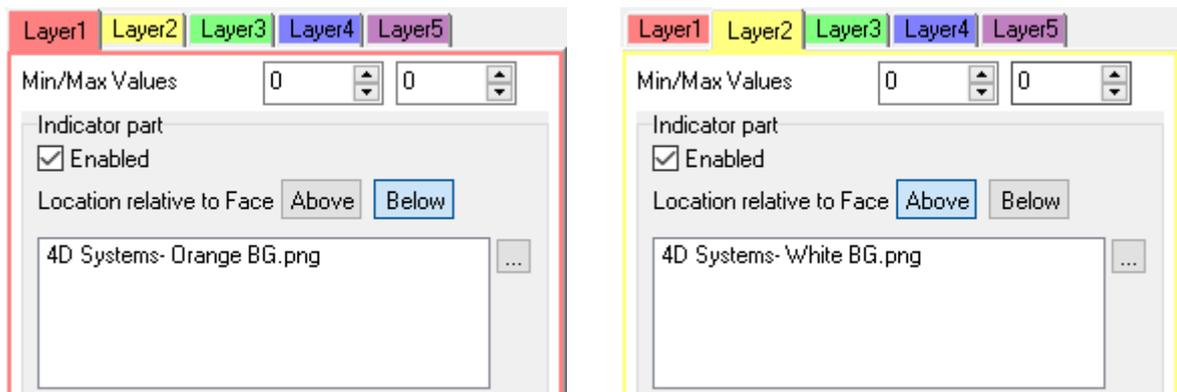
Notice that Layer 1 (logo with white background) covers small portions of the face image (hollow rectangle) and Layer 2 (logo with orange background). While the logo with orange background is covered by both the face image and Layer 1.



Pressing the down arrow of a layer in the layer guide section will swap its contents with the adjacent layer below it. Pressing the up arrow of a layer in the layer guide section will swap its contents with the adjacent layer above it. To swap the contents of Layer 1 and Layer 2, click any of the two buttons indicated in the layer guide below.



By doing so, the contents of the layers will change



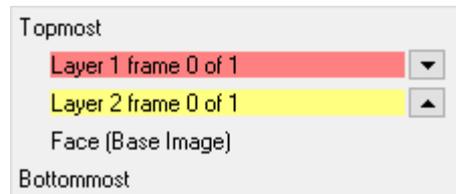
However, if you look carefully, the logo with orange background is still below the base image while the logo with the white background is still above the base image. Therefore, the preview area will still show the same.



Therefore, Layer 1 which previously contains an image (orange) above the face image now has an image (white) below the face image while Layer 2 which previously contains an image (white) below the face image now has an image (orange) above the face image.

As shown in the previous discussion, the contents of both layers were swapped. But it doesn't really change the appearance in the preview area since swapping includes all the parameters of each layer including the layer's position with respect to the face image.

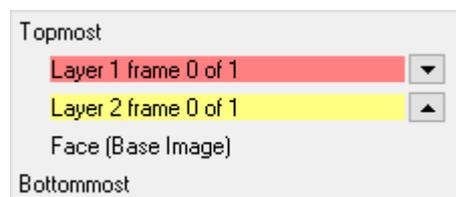
Swapping layers that are both above or both below is a different case.



Using same layers as the previous example but this time both are above the face image.



By clicking the swap buttons, it can be noticed that Layer 1 is still above Layer 2.



The reason behind this is discussed in the previous section, *Order of Layers*.

Even if Layer 1 is still above Layer2, their contents have been swapped. Thus it can be seen that the logo with orange background covers some portion of the logo with white background.



## 7. Setting Layer Parameters

Since a layer is used to display a single manipulated image, a series of static images or a series of manipulated images, it can have one or more frames. Furthermore, each frame is associated to a value of that layer. The number of frames in a layer is reflected in the **Min/Max Values** parameters.



The total number of frames is specified by the user by setting the values of the **Min/Max Values** parameters.

For a layer with Numerical Part enabled and set to String Lookup, the number of frames corresponds to the number of Strings added to the image list. For these cases, therefore, the **Min/Max Values** parameter values are automatically set by the Smart Widgets Editor tool as soon as the string items are added.

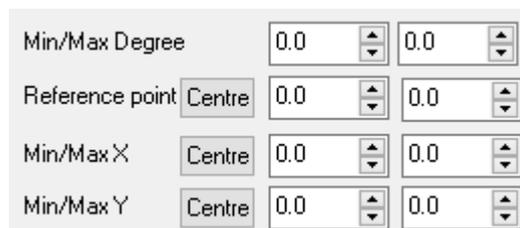
**Note:** All methods of manipulating images in a layer are associated with the **Min/Max Values**. Manipulating images in a layer requires that the **Min/Max Values** cover multiple frames.

### 7.2. Layers with a Manipulated Image

Images contained in each layer can be manipulated to move horizontally, vertically and/or rotate at certain angles.

This section will discuss how to manipulate a single image in a layer.

An image can be manipulated by setting the values of the parameters shown in the image below.



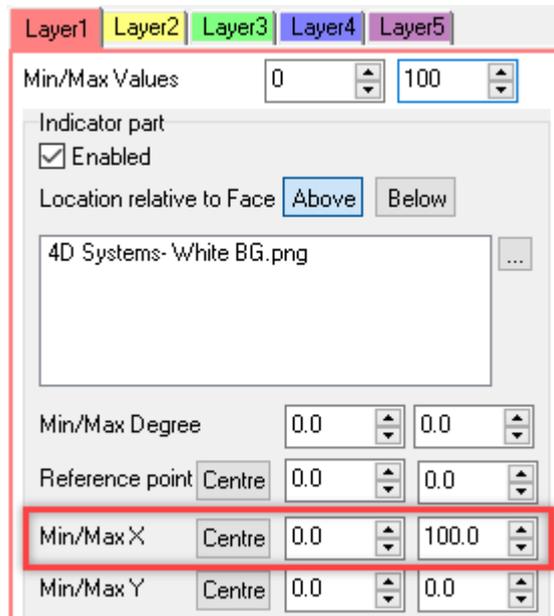
To move the image along the x and/or y axes, set the values of parameters **Min/Max X** and/or **Min/Max Y**. To rotate the image at certain angles, set the values of the parameters **Min/Max Degree**. To specify the point of rotation of the image, set the values of the parameter **Reference point**. These are all in conjunction with **Min/Max Values**.

These parameters and their corresponding manipulation effects are discussed in detail in the subsequent sections. By using a combination of different manipulation parameters, it is possible to create more complex motions for any images.

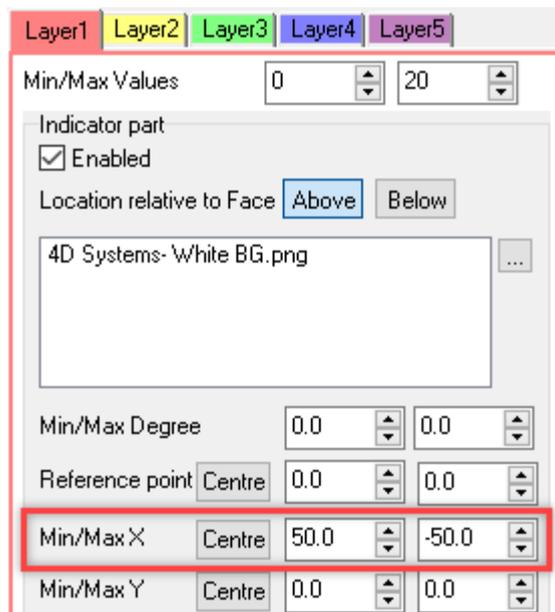
#### 7.2.1. Horizontal Motion

A layer may contain an image that moves horizontally by the utilizing the X positions for the minimum and maximum layer values (**Min/Max X**). The tool moves the manipulated image from left to right or right to left depending on which is greater between the X positions of the maximum value and the minimum value.

The **Min/Max X** position values serve as the end positions of the reference point of the image in the working area. By default, the reference point of an image is its top-left corner (0, 0).

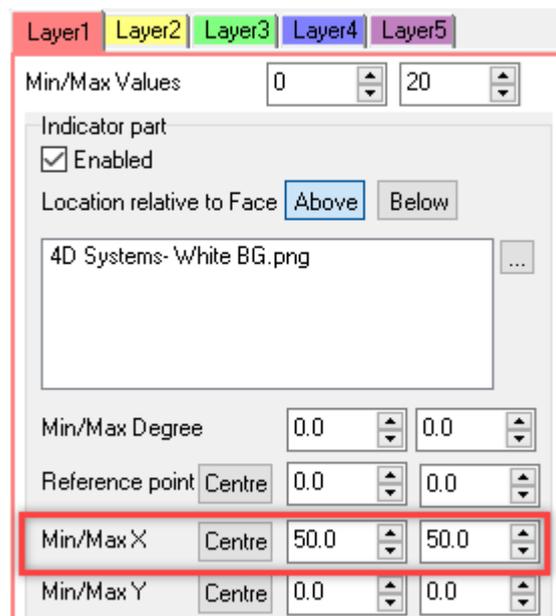


In this example, **4D Systems- White BG.png** is moved left to right from 0 to 100 X positions from values 0 to 100.



While in this example, **4D Systems- White BG.png** is moved right to left from 50 to -50 X positions from values 0 to 20.

By default, both X position values are set to zero to have no horizontal motions. By using equal values for the X position, the tool prevents horizontal motions. Furthermore, the image will always be displayed with its reference point at the set X position.

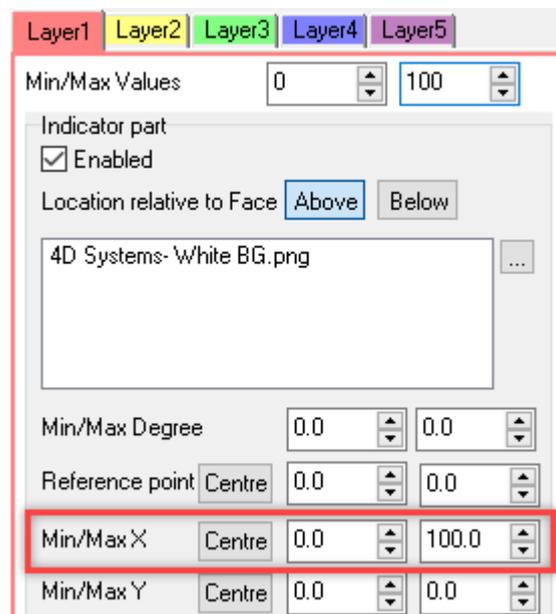


In this example, **4D Systems- White BG.png** is displayed at 50 X position from values 0 to 20.

Here's a simple demonstration on what to expect.

**Min and Max Values:** 0 to 100

**Min/Max X:** 0 to 100



At Frame 0:



At Frame 50:



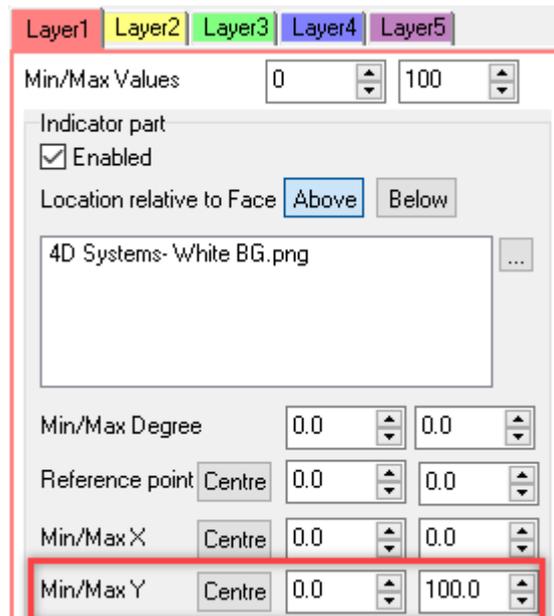
At Frame 100:



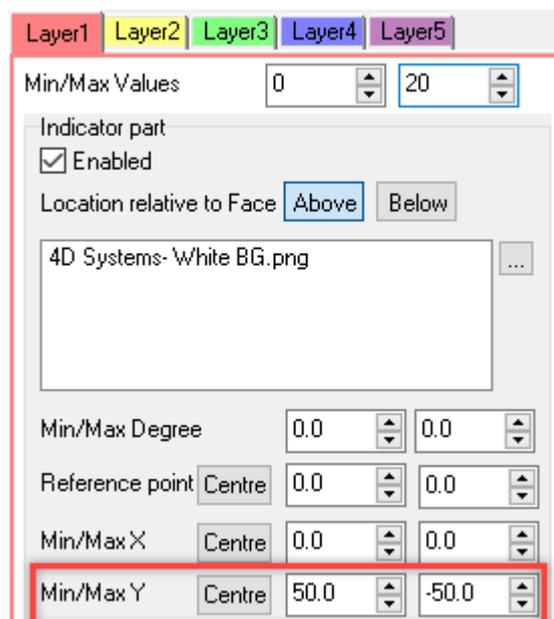
### 7.2.2. Vertical Motion

A layer may contain an image that can be manipulated to move vertically by utilizing the Y positions for the minimum and maximum layer values (**Min/Max Y**). The tool moves the manipulated image downwards or upwards depending on which is greater between the Y positions of the maximum value and of the minimum value.

The **Min/Max Y** position values serve as end positions for the reference point of the image in the working area. By default, the reference point of an image is its top-left corner (0, 0).

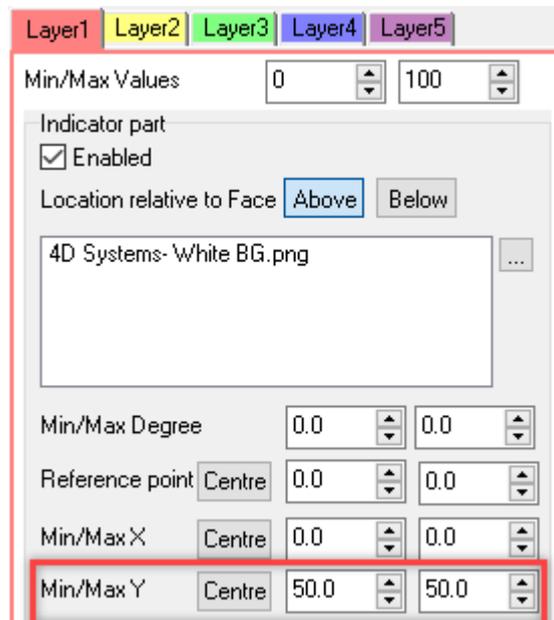


In this example, **4D Systems-White BG.png** is moved downwards from 0 to 100 Y positions from the values 0 to 100.



While in this example, **4D Systems-White BG.png** is moved upwards from 50 to -50 Y positions from values 0 to 20.

By default, both Y position values are set to zero to have no vertical motions. By using equal values for the Y position, the tool prevents vertical motions. Furthermore, the image will always be displayed with its rotation point at the set Y position.



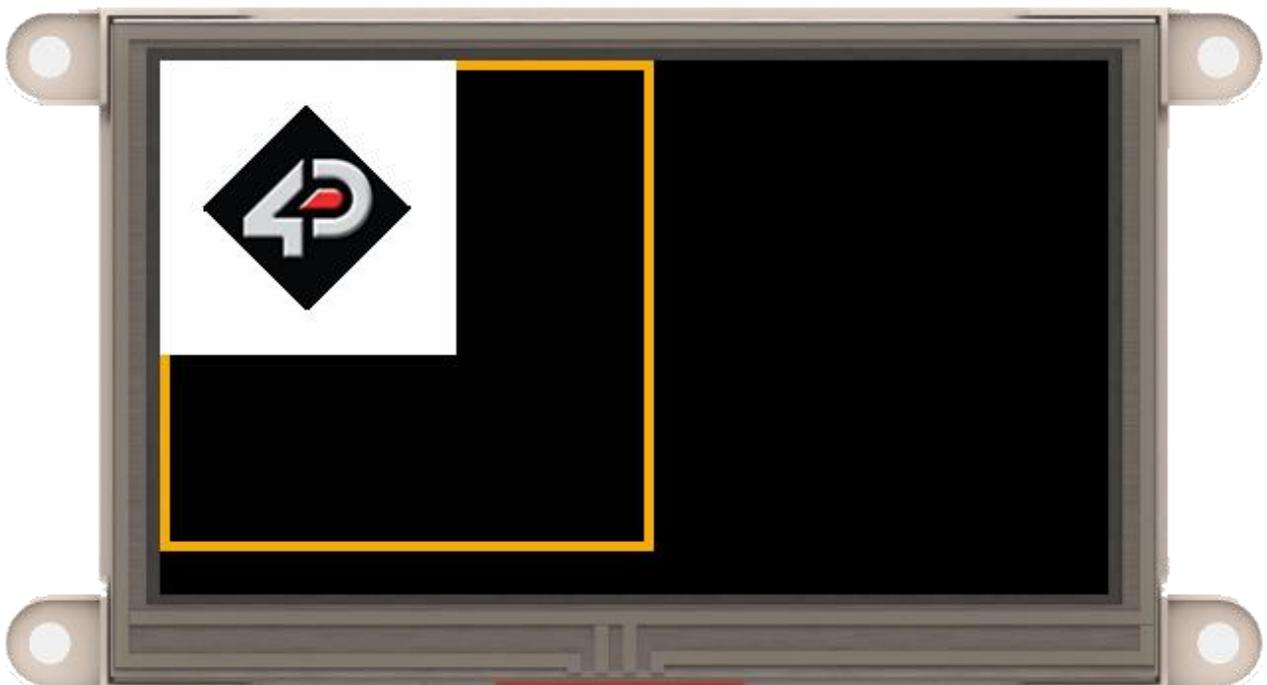
In this example, **4D Systems-White BG.png** is displayed at 50 Y position from values 0 to 100.

Here's a simple demonstration on what to expect.

**Min and Max Values:** 0 to 100

**Min/Max Y:** 0 to 100

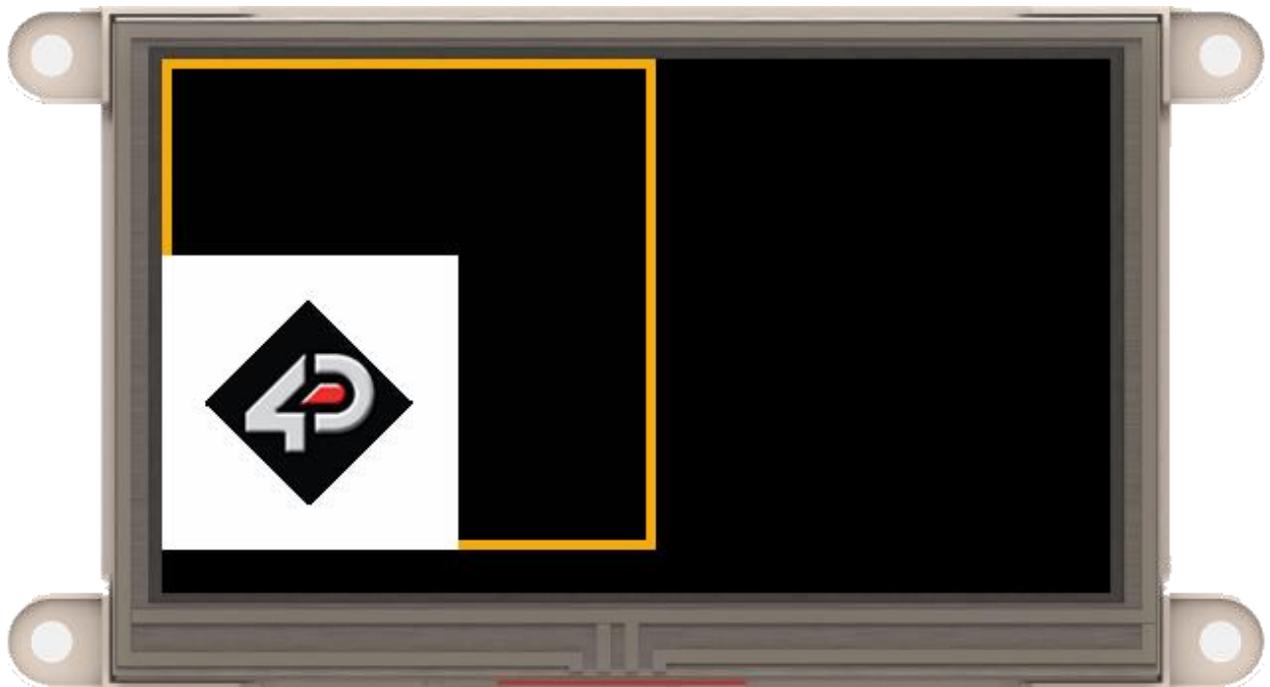
At Frame 0:



At Frame 50:

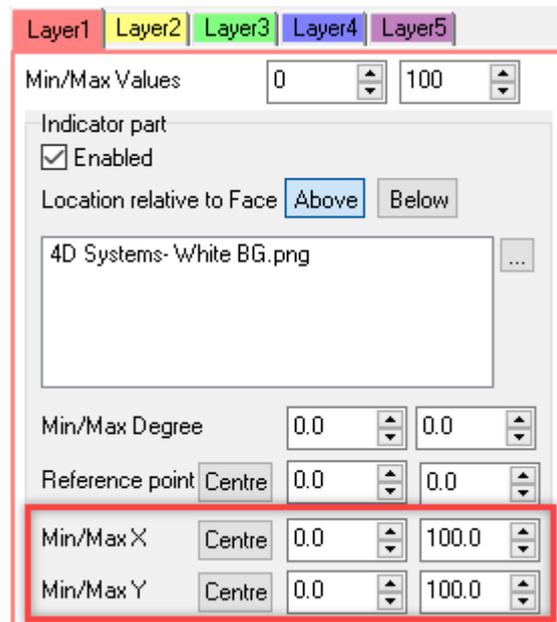


At Frame 100:



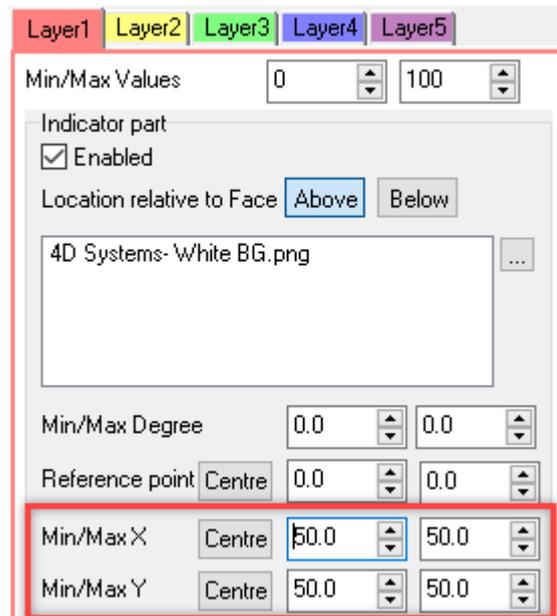
### 7.2.3. Diagonal Motion

By combining **Horizontal Motion** and **Vertical Motion**, it is possible to move a layer image diagonally.



In this example, with 101 frames, **4D Systems-White BG.png** moves diagonally from **(0, 0)** to **(100, 100)**.

As discussed in **Horizontal Motion** and **Vertical Motion**, setting the minimum and maximum positions for any of the two (**Min/Max X** and **Min/Max Y**) prevents motion in the respective directions. Therefore, setting both to have equal values will give the image a fixed position.



In this example, **4D Systems-White BG.png** is located at (50, 50). That means the reference point, which is the top-left corner (0, 0) of the image (as seen on the image above), is located at (50, 50) of the working area. It is really important to note that the X and Y positions correspond to the position of the **Reference Point** of the image with respect to the **working area** rather than **preview area**.

Here's a simple demonstration on what to expect.

**Min and Max Values:** 0 to 100

**Min/Max X:** 0 to 100

**Min/Max Y:** 0 to 100

At Frame 0:



At Frame 50:



At Frame 100:

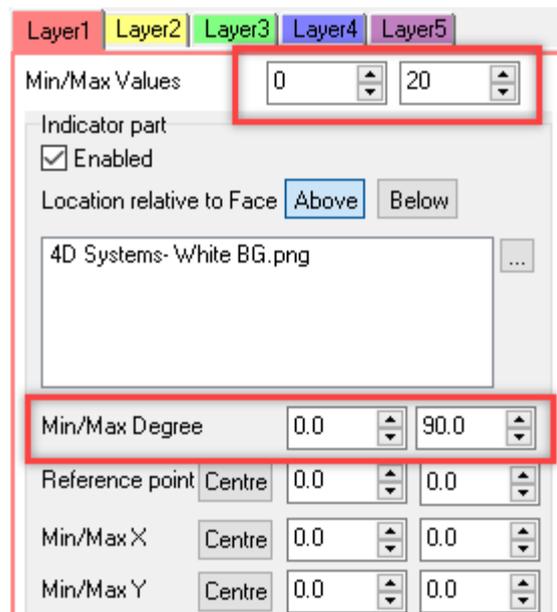


### 7.2.4. Angular Motion

The image on a layer can also be manipulated to rotate at certain angles by the varying the values of the parameters **Min/Max Degree**. Smart Widgets Editor rotates the image clockwise or counter-clockwise depending on the values of the **Min/Max Degree** parameters.

The **Minimum Degree** parameter is the angle of the image at its starting position while the **Maximum Degree** parameter is the angle of the image at its ending position. A negative angle rotates the image counter-clockwise, while a positive value rotates the image clockwise.

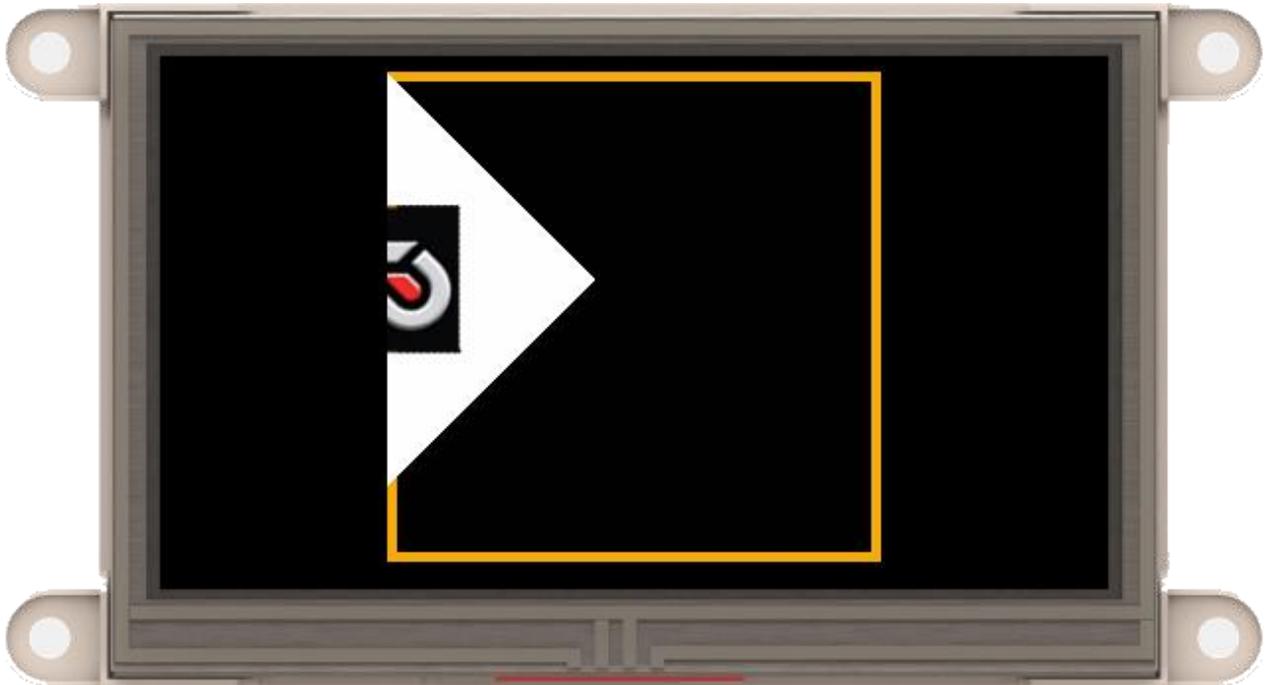
In the example shown below, the image has 21 frames, indexed **0** to **20**, rotating 90 degrees clockwise.



At frame 0, the image is at its original position.



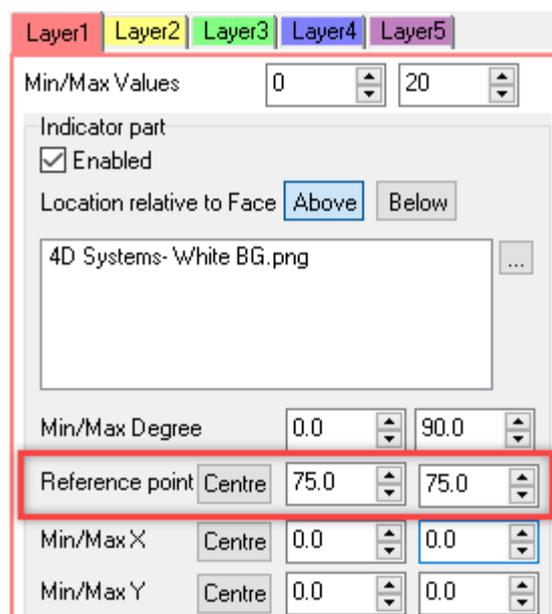
At frame 10, the image is rotated by 45 degrees clockwise.



Notice that the image was rotated at its top-left corner  $(0, 0)$ , which is the default reference point. The values of the parameter **Reference Point** specify the coordinates of the rotation point of an image. The coordinates of the reference point are relative to the top-left corner of the image. Again, the reference point is at its top-left corner by default – point  $(0, 0)$ .

**Note:** The part of the logo outside of the working area is not shown. This is true for all image manipulation methods. The Smart Widget will only take the set **Working Area** which is the area of the face image by default. For more information, refer to **Size of the Working Area**.

The coordinate values below map the reference point of the logo to its centre.



Furthermore, as mentioned in the previous sections, the **Min/Max X** and **Y** position values serve as end positions for the reference point of the image in the working area.

Thus the parameters shown above will give this output.



As discussed in the previous sections, setting the parameter **Min X** and **Y** and **Max X** and **Y** to equal values will give the image a fixed position in the **working area**.

In the example below, the reference point of the image will remain at point **(125, 125)** of the **working area** at all frames.

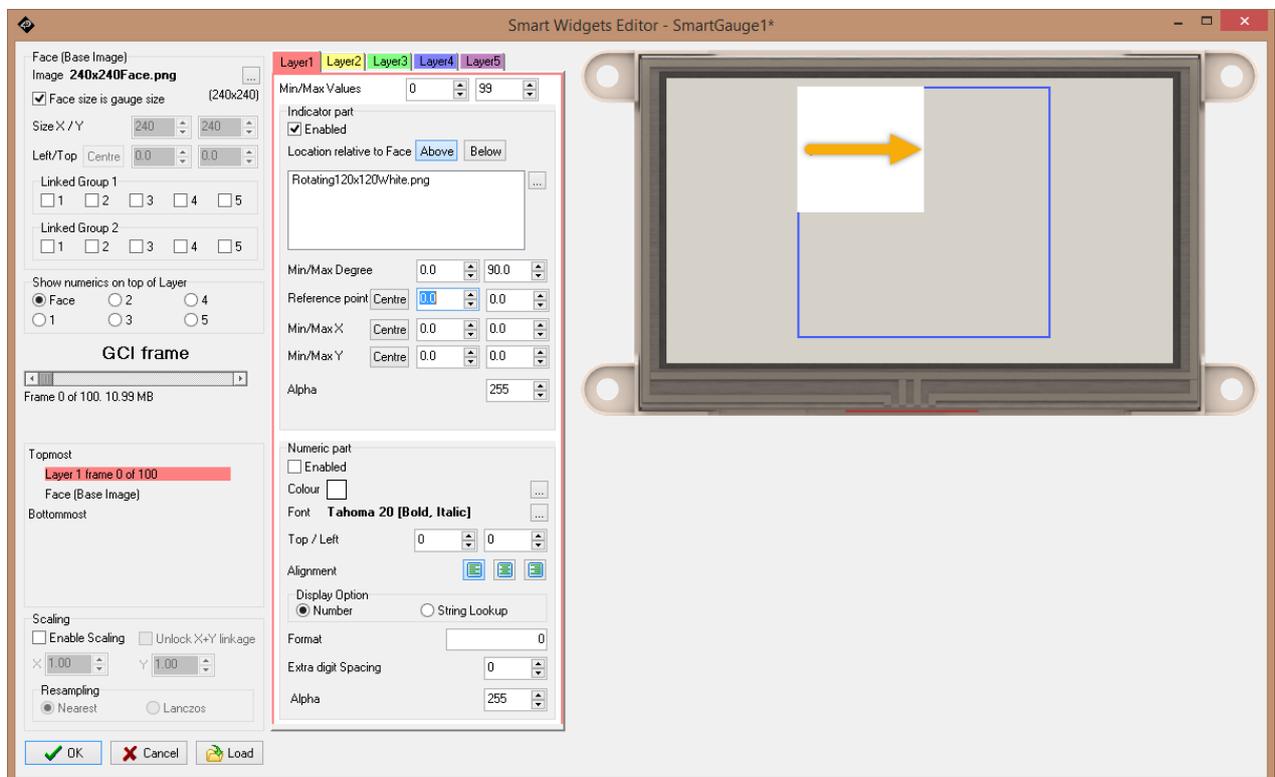
Layer1	Layer2	Layer3	Layer4	Layer5
Min/Max Values	0		20	
Indicator part				
	<input checked="" type="checkbox"/>	Enabled		
Location relative to Face	Above	Below		
	4D Systems- White BG.png			...
Min/Max Degree	0.0		90.0	
Reference point	Centre	75.0	75.0	
Min/Max X	Centre	125.0	125.0	
Min/Max Y	Centre	125.0	125.0	
Alpha	255			

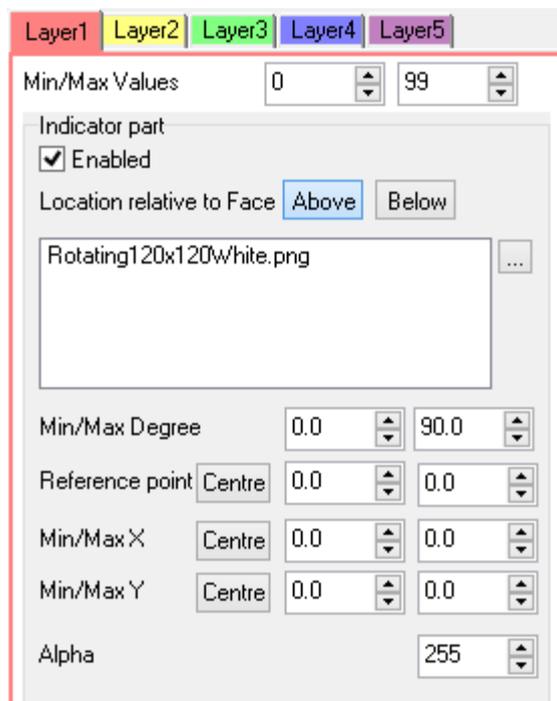
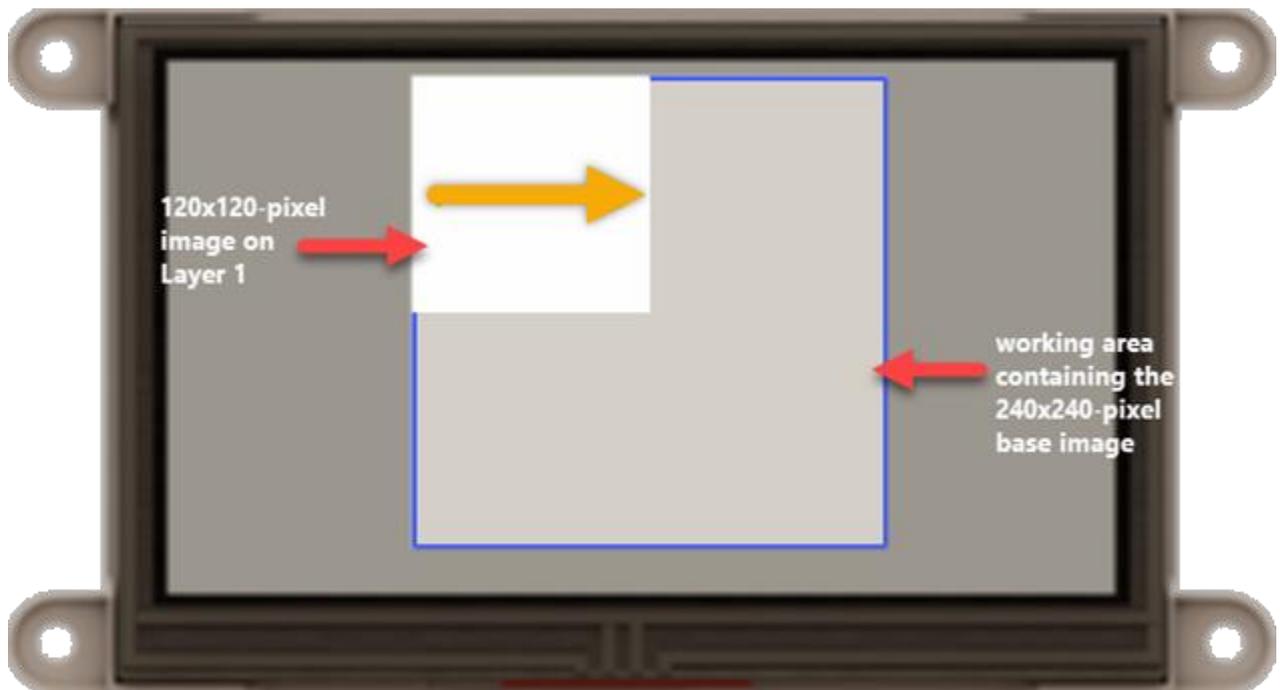
Here's the output:



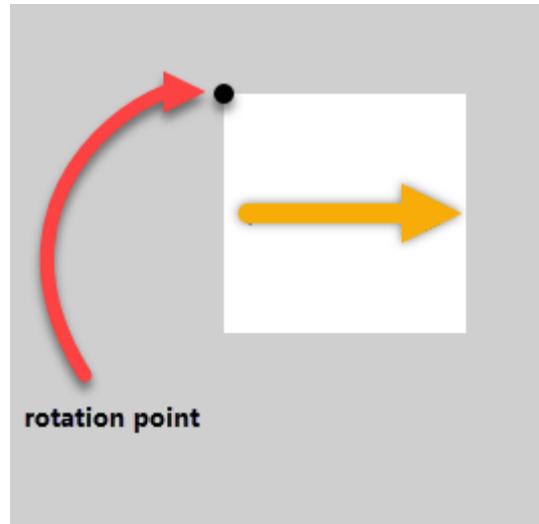
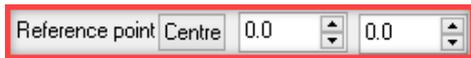
The following paragraphs discuss the use of the angular motion parameters in more detail using an actual example.

Consider the image on Layer 1 in the example below.

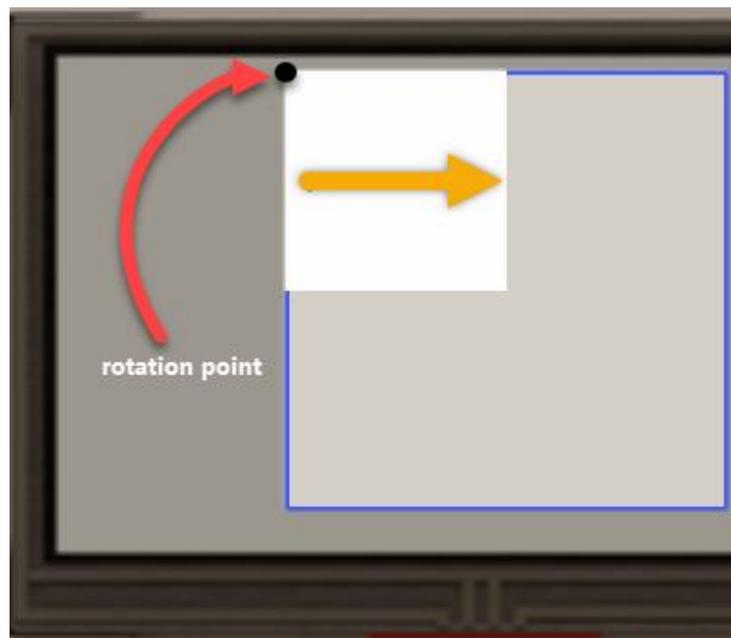
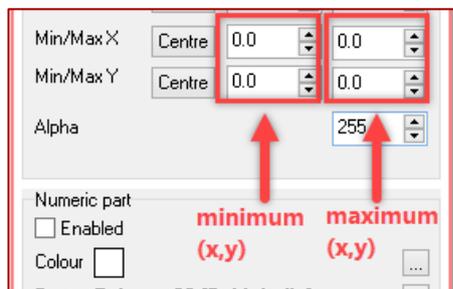




Layer 1 has 100 frames, indexed from 0 to 99. Note also that the Minimum Degree and Maximum Degree parameters are set to 0 and 90, respectively. This means that at frame 0, the image on Layer 1 is at its original position. At frame 99, the image on Layer 1 will have rotated by 90 degrees, clockwise. The point of rotation of the image is at point **(0,0)**, relative to its top-left corner.

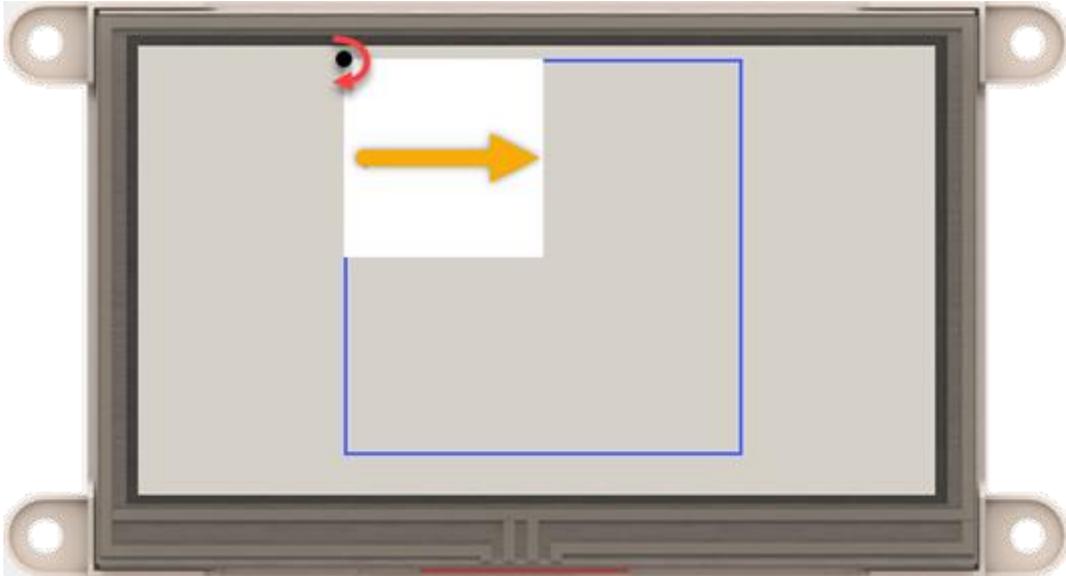


The tool then places the image on the working area.

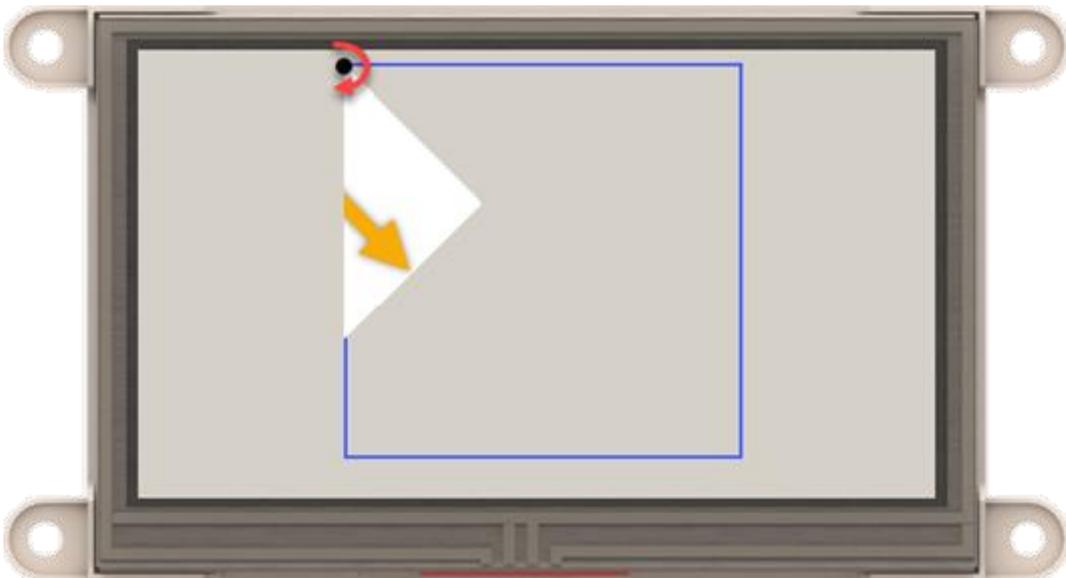


Based on the parameter values above, at frame 0, the rotation point of the image coincides with point **(0, 0)** of the working area (**Min X** and **Min Y**). At frame 99, the rotation point of the image still coincides with point **(0, 0)** of the working area (**Max X** and **Max Y**). Thus, the image remains at a fixed position at all frames.

At frame 0:

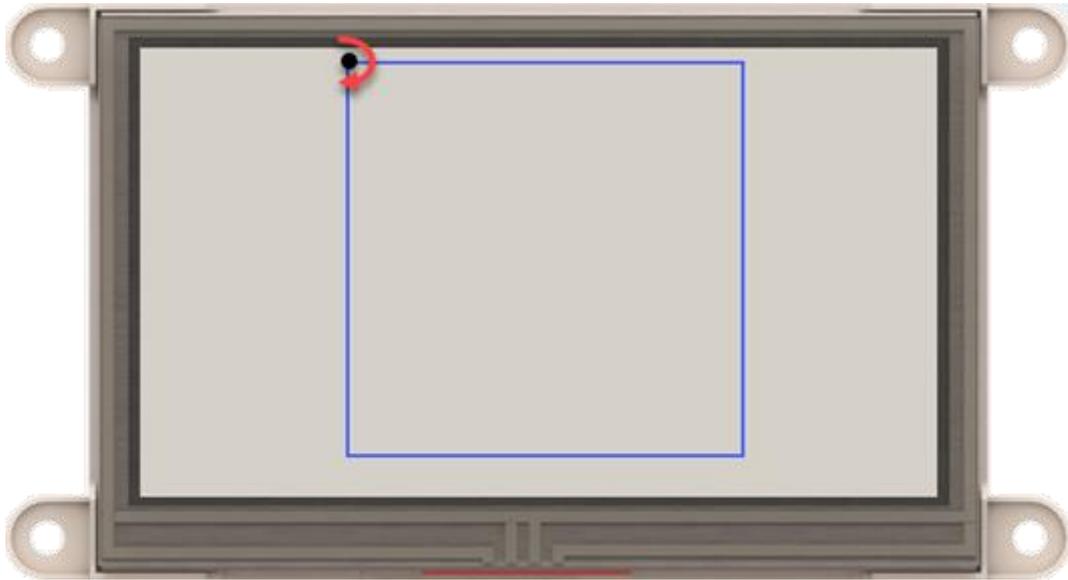


At frame 50:



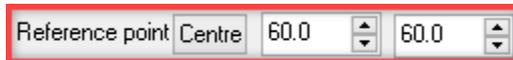
**Note:** The part of the logo outside of the working area is not shown. This is true for all image manipulation methods. The Smart Widget will only take the set Working Area which is the area of the face image by default. For more information, refer to Size of the Working Area.

At frame 99:

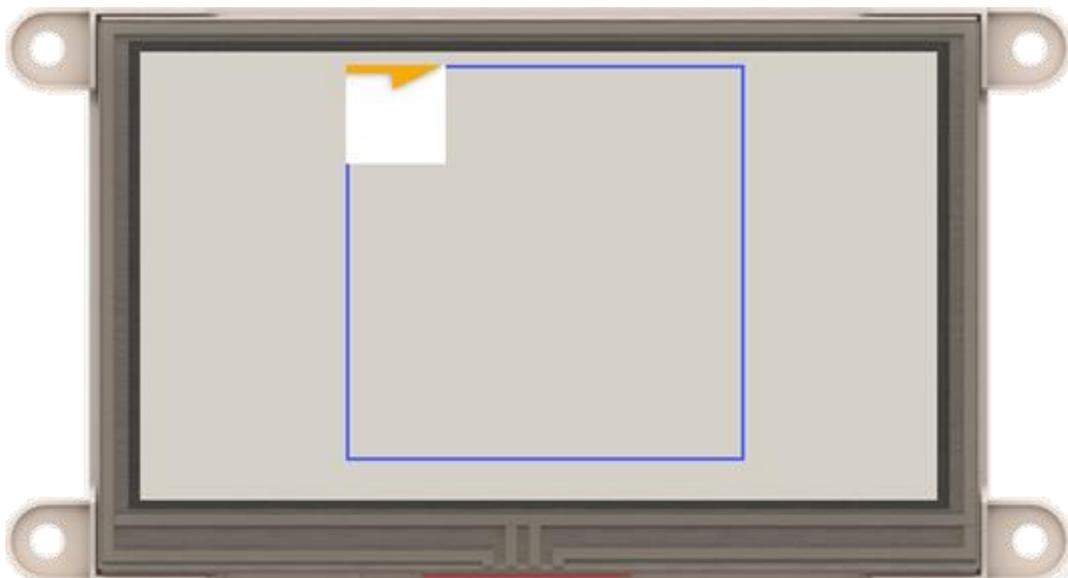


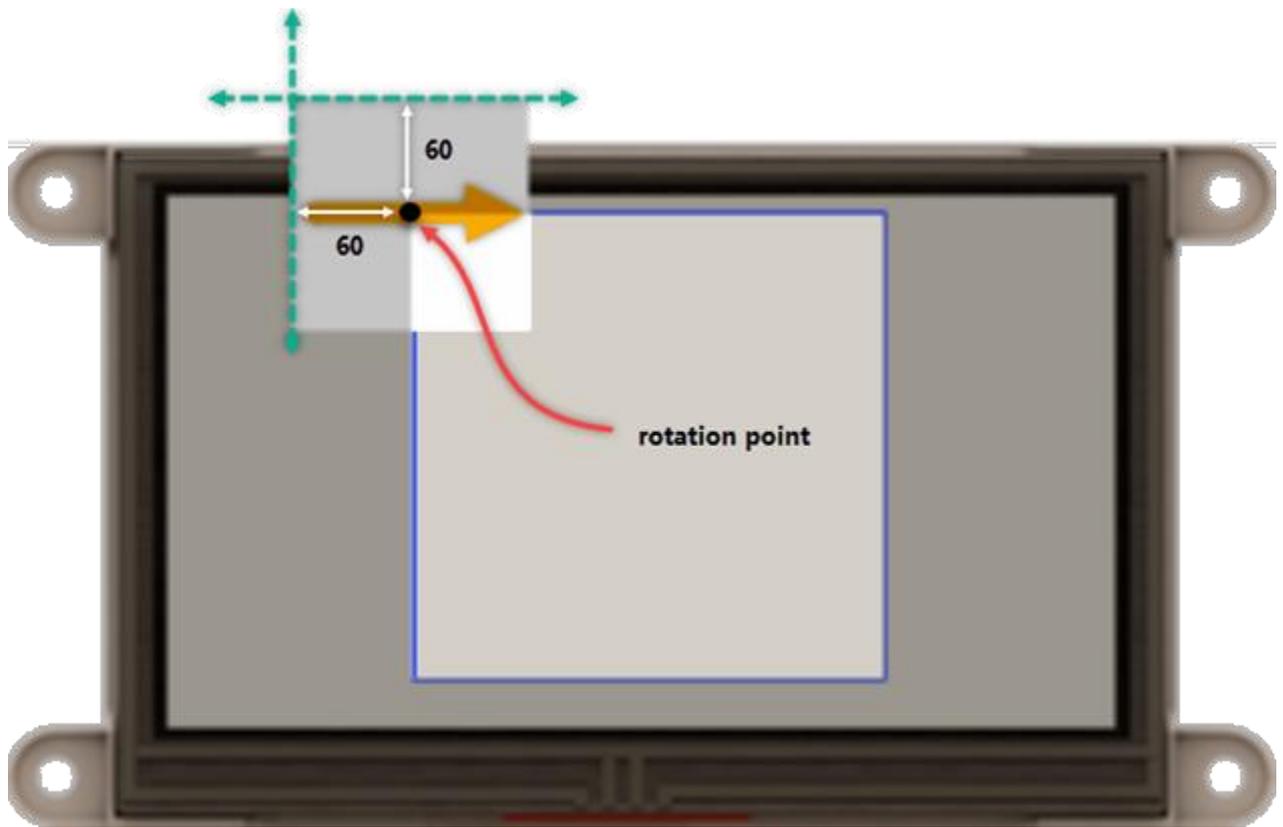
**Note:** The part of the logo outside of the working area is not shown. This is true for all image manipulation methods. The Smart Widget will only take the set Working Area which is the area of the face image by default. For more information, refer to Size of the Working Area.

To demonstrate how the rotation point is changed, the values of the **Reference point** parameters are now set to the values shown below.

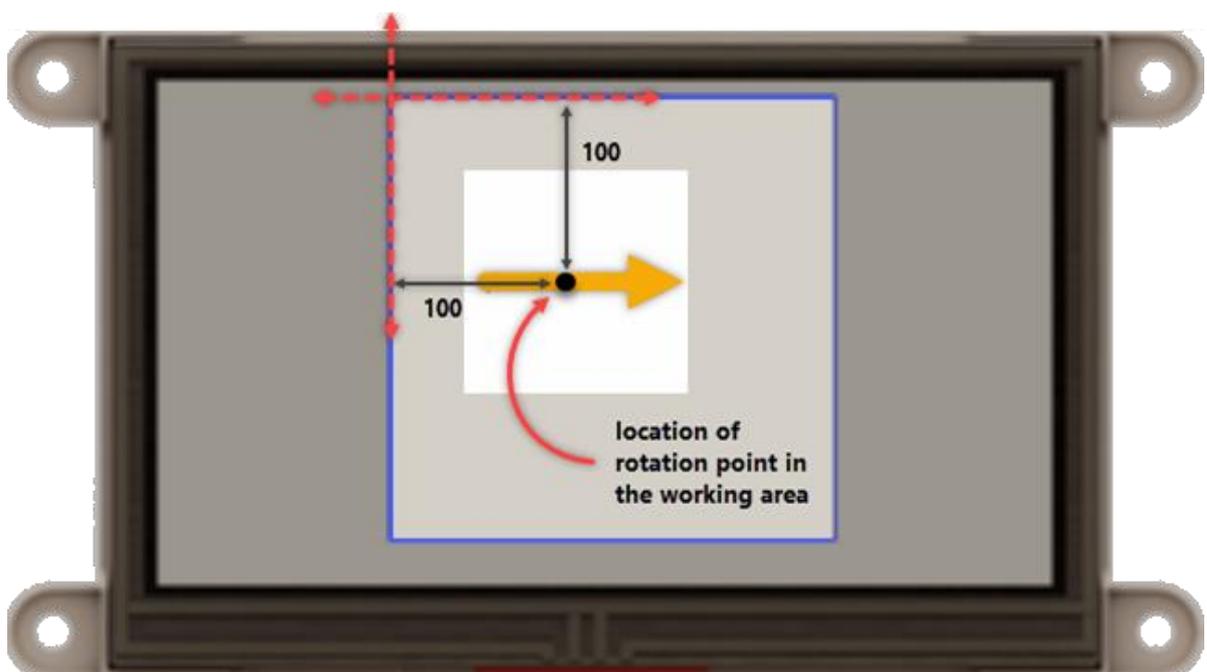
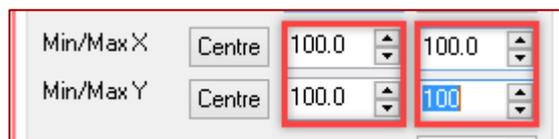


The rotation point of the 120x120-pixel image is now at its center. Note however that only a quarter of the image is actually visible in the working area.



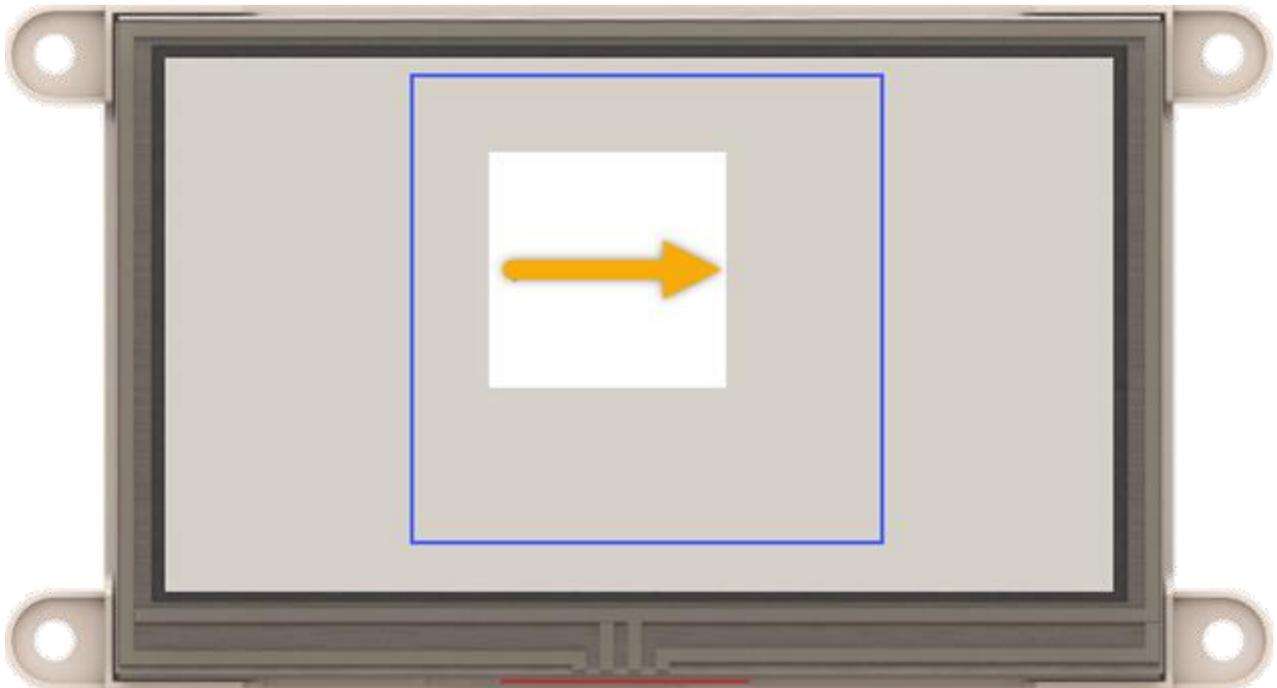


For the entire image to be visible in the working area at frame 0, the location of its rotation point is moved to point **(100, 100)** on the working area.

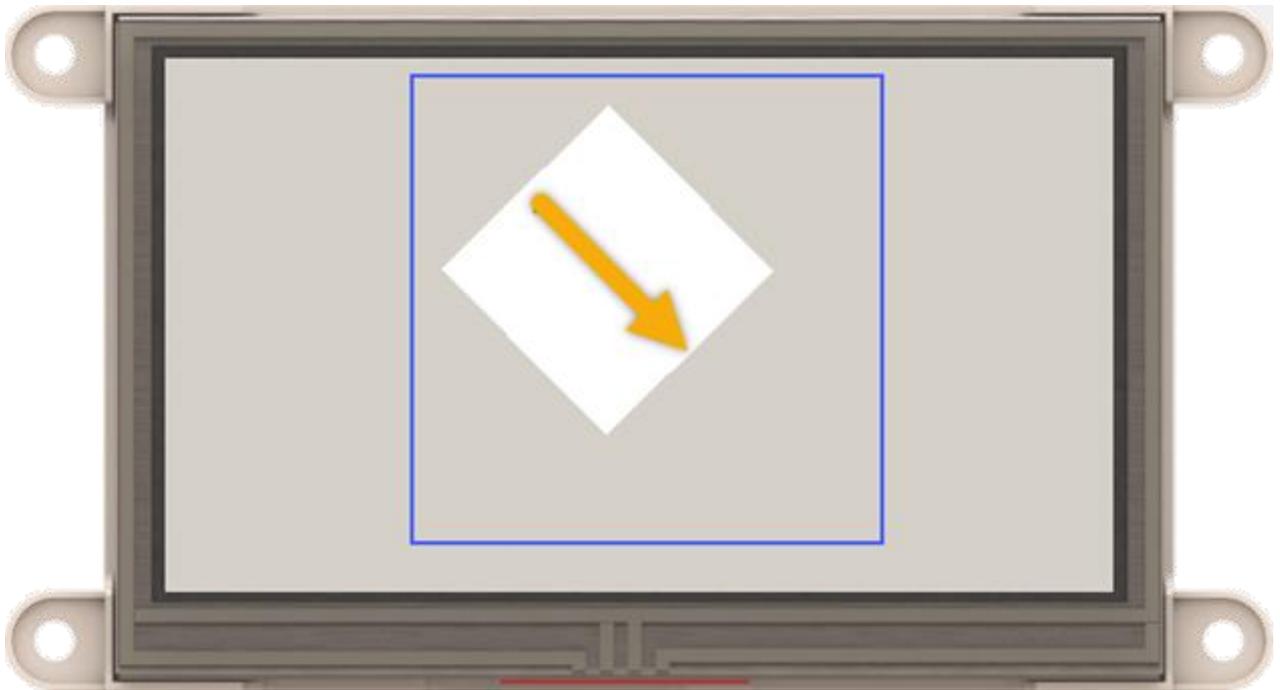


Based on the new **Min/Max X** and **Y** parameters above, the rotation point of the image will remain fixed at point **(100, 100)** on the working area at all frames.

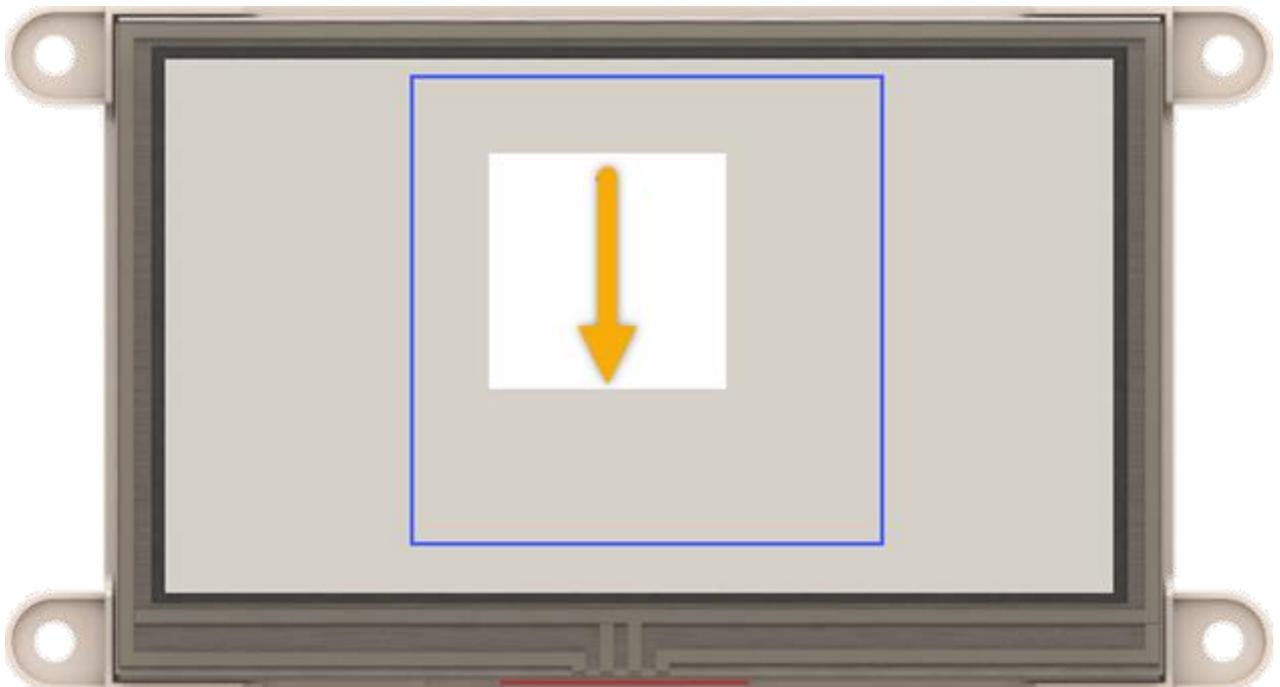
At frame 0:



At frame 50:

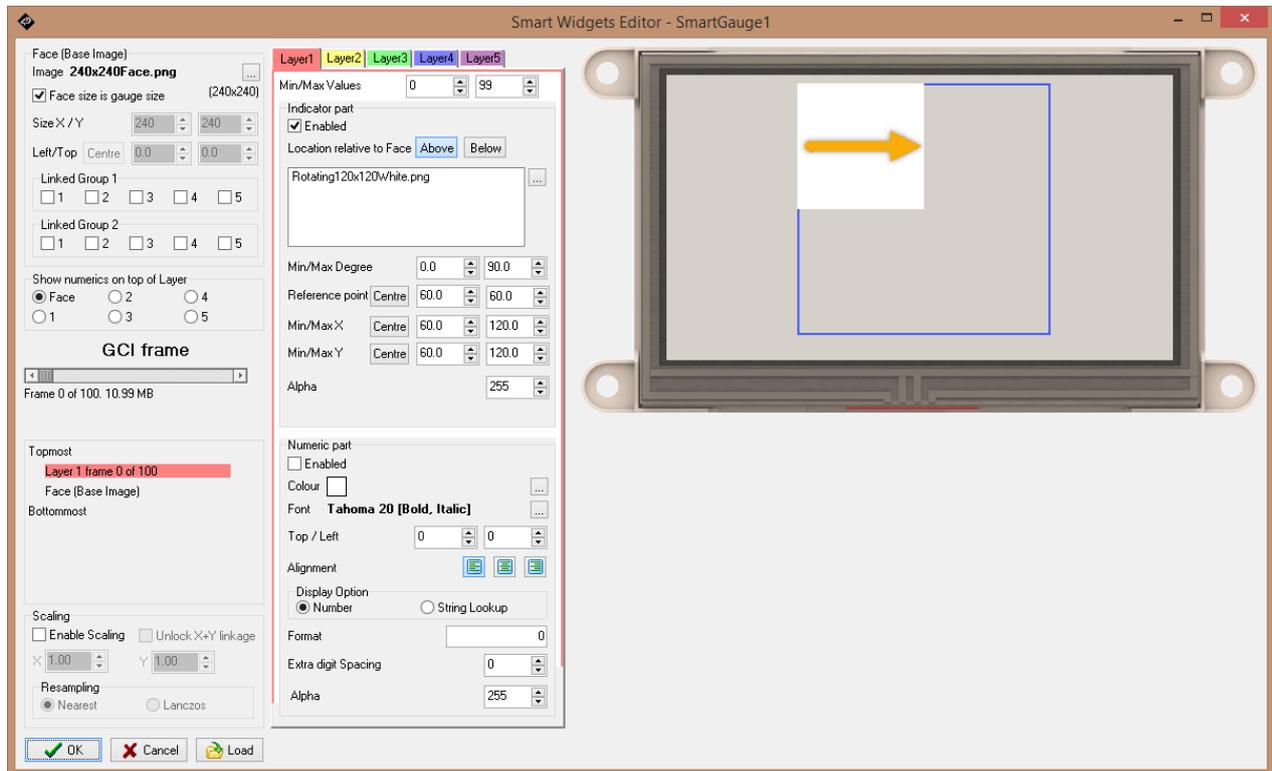


At frame 99:



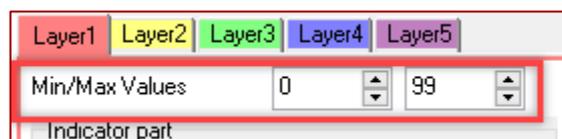
### 7.2.5. Combination of Diagonal and Angular Motions

By properly configuring the manipulation parameters, it is possible to move an image along the x and y axes while rotating it at the same time. The animation of such object involves all the three basic motions (horizontal, vertical, and angular). Consider the example shown below, which is similar to the example used in the previous section ([Angular Motion](#)).

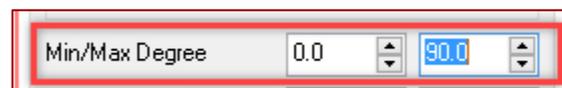


The image on Layer 1 is a 120x120-pixel yellow arrow with a white background.

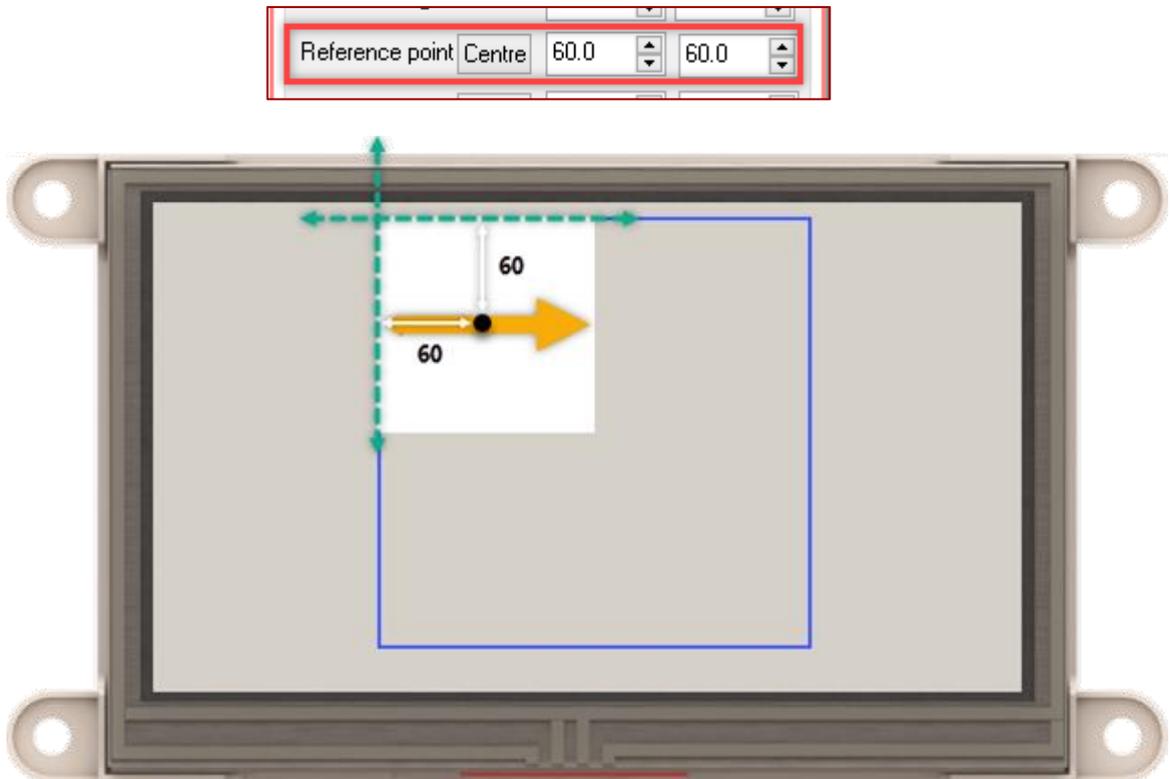
The **Min/Max Values** parameter values indicate that Layer 1 has 100 frames, indexed from 0 to 99.



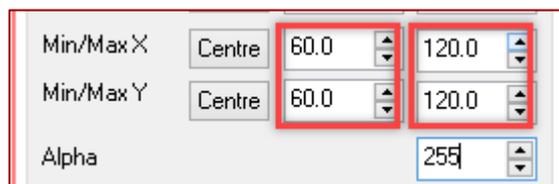
The **Min/Max Degree** parameter values indicate that the image will be rotated clockwise by 90 degrees from its original position.



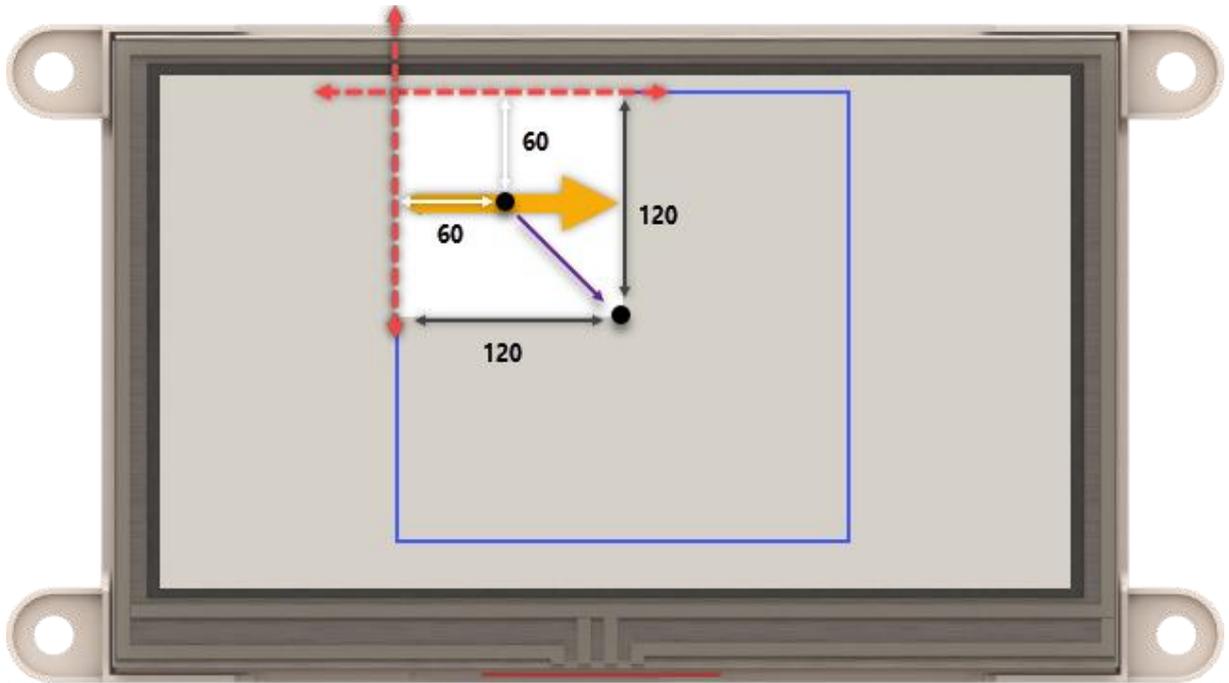
The **Reference point** parameter values indicate that the rotation point of the image is at its center.



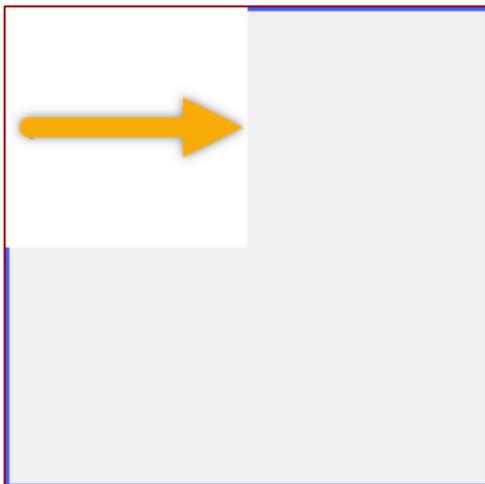
The **Min/Max X** and **Y** parameter values indicate that the image, by its center (point of rotation), will be moved from point **(60, 60)** to point **(120, 120)** on the working area from frame 0 to frame 99.



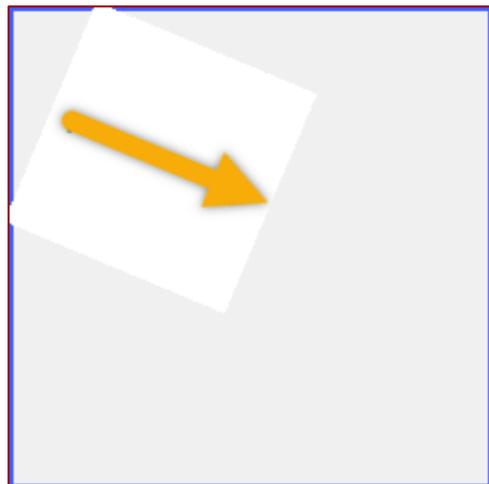
The purple arrow in the image below shows the path of the rotation point as it diagonally moves across the working area.



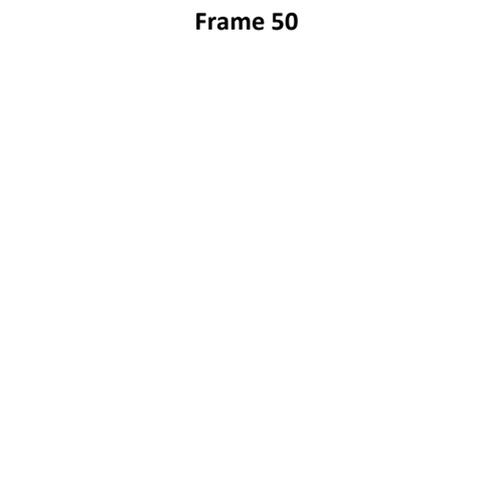
Frame 0



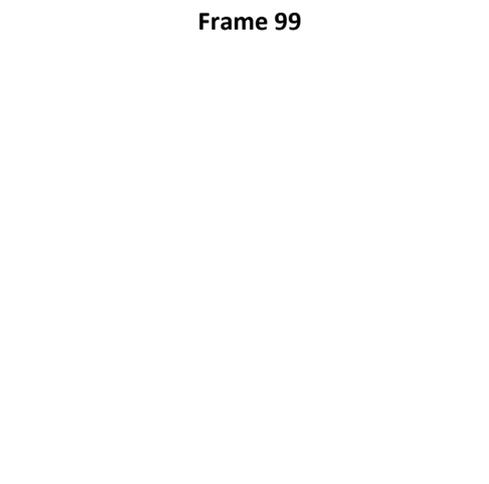
Frame 25

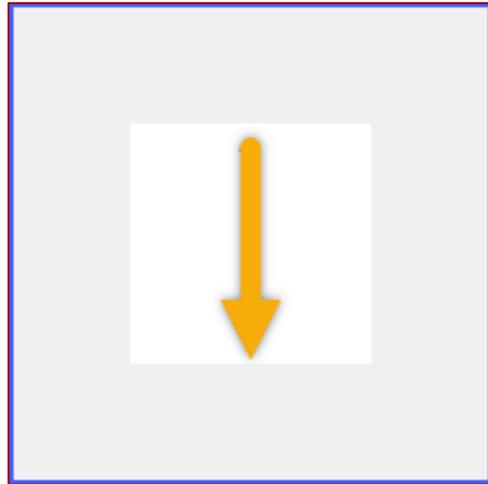
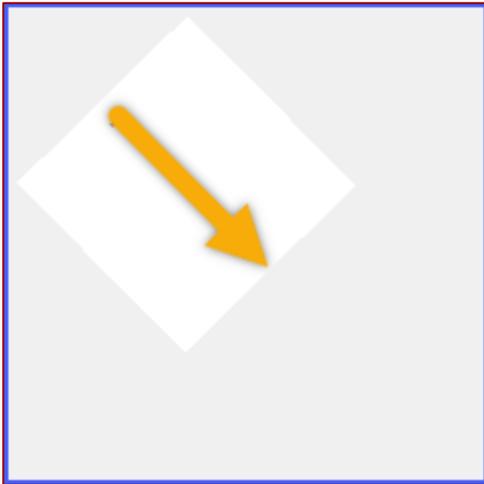


Frame 50



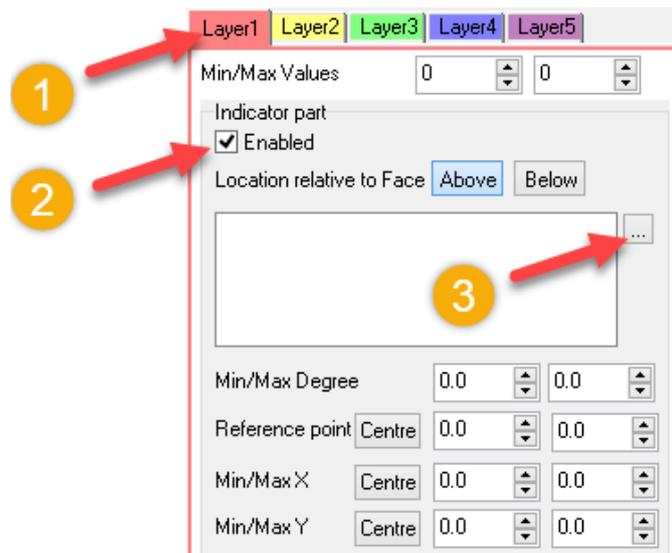
Frame 99



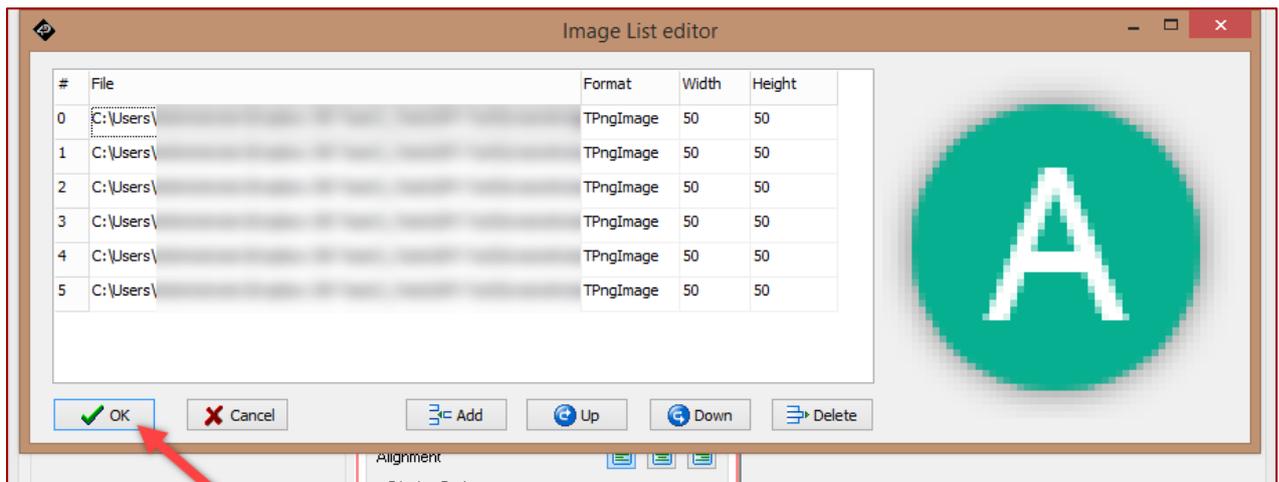
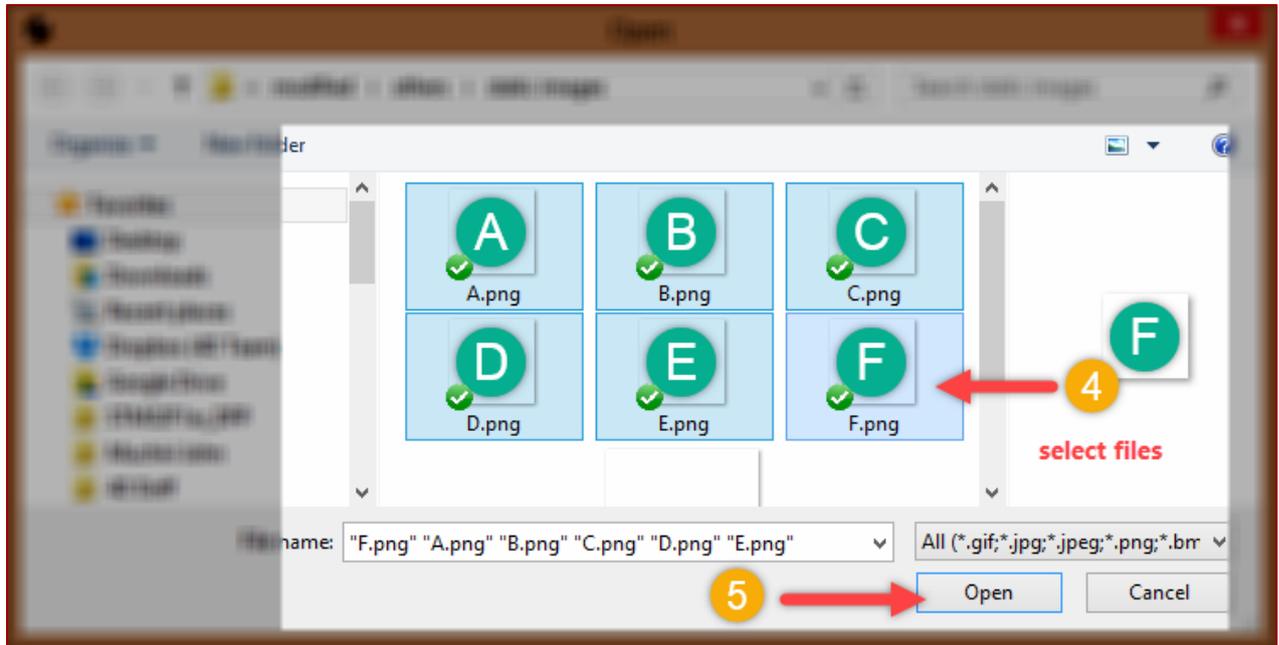


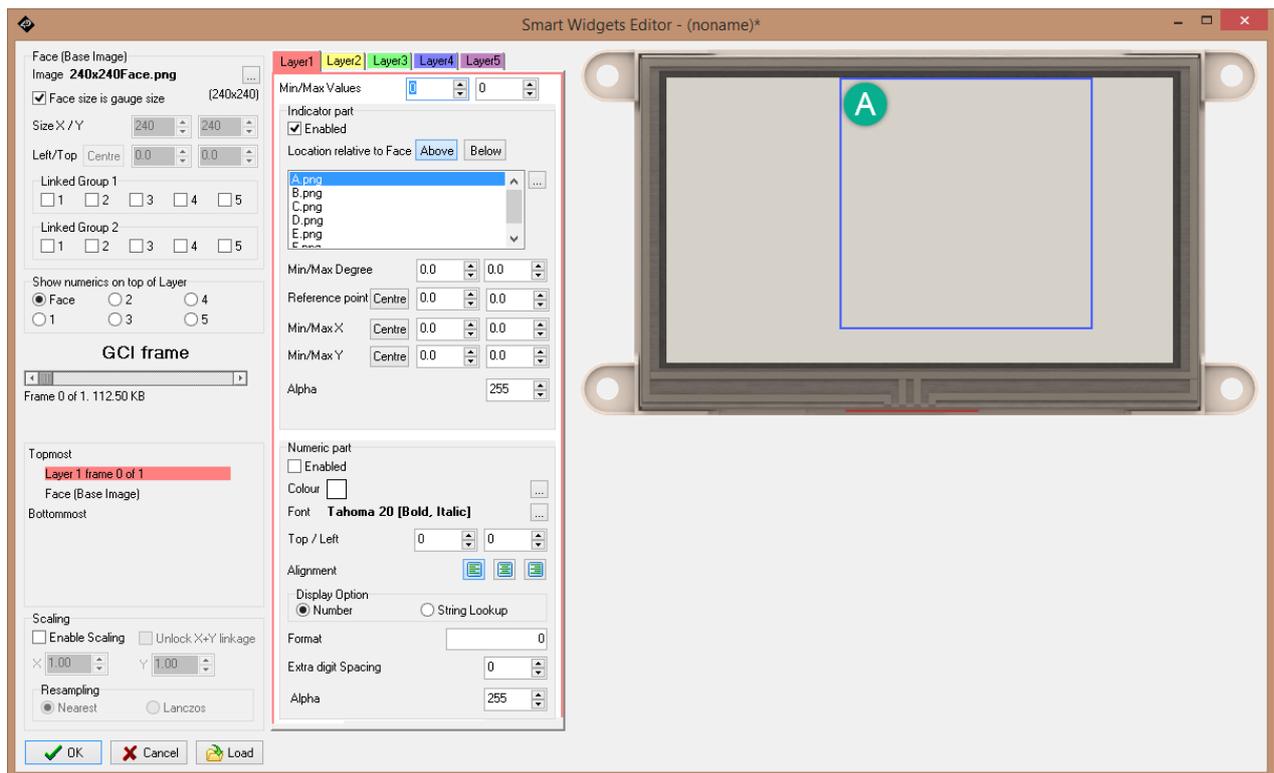
### 7.3. Layers with Multiple Fixed Images

This section discusses about layers containing multiple fixed images. It is possible to make a layer show a series of fixed, non-moving or static images. The procedure for doing this is shown below.



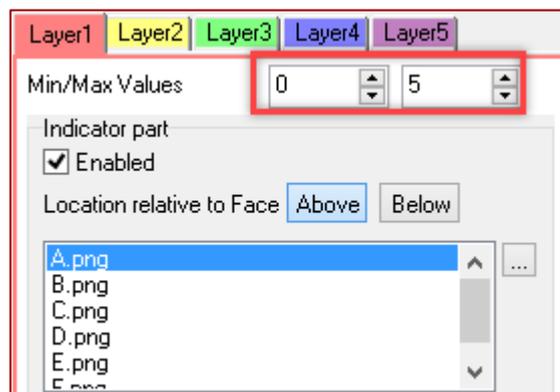
In this example, PNG images containing a letter are used.





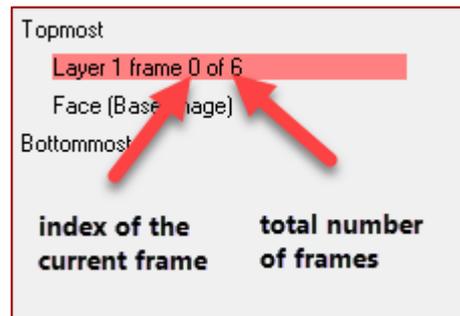
### 7.3.1. Frames and Values

Set the Min and Max Values to 0 and 5, respectively. This results to a total of 6 frames, enough to represent the six images.

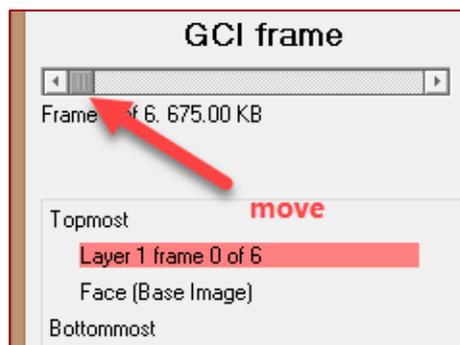


### 7.3.2. Display the Static Images

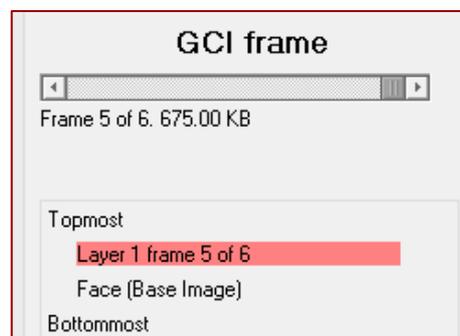
The generated frames of a layer containing static images are indexed from **0** up to ***N-1***, where ***N*** is the total number of frames. The first frame shown in the working area is the first file in the image list when the static images were added. Note also that, in the layer guide, the index of the current frame and the total number of frames are shown. The image below shows the index of the current frame and the total number of frames on Layer1.

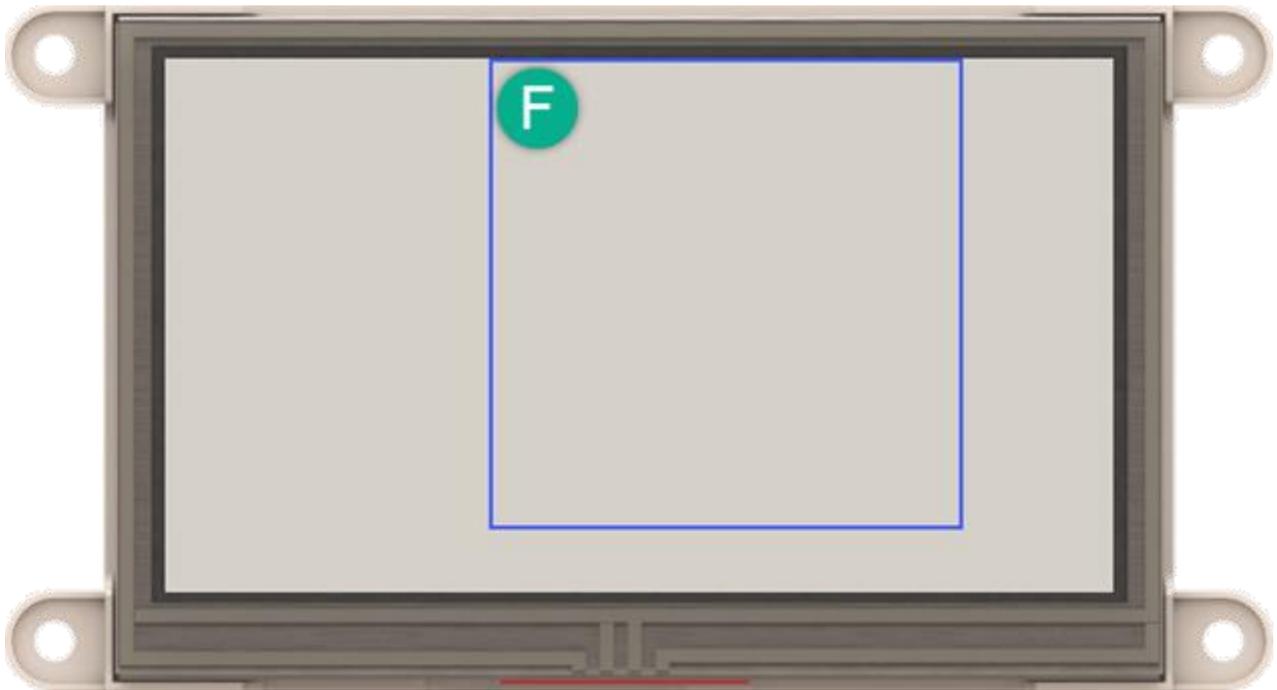


To show the rest of the static images, move the thumb of the GCI frame slider to the right.



The image below shows the thumb positioned at the end of the slider. Hence, the last static image is currently displayed on the working area.





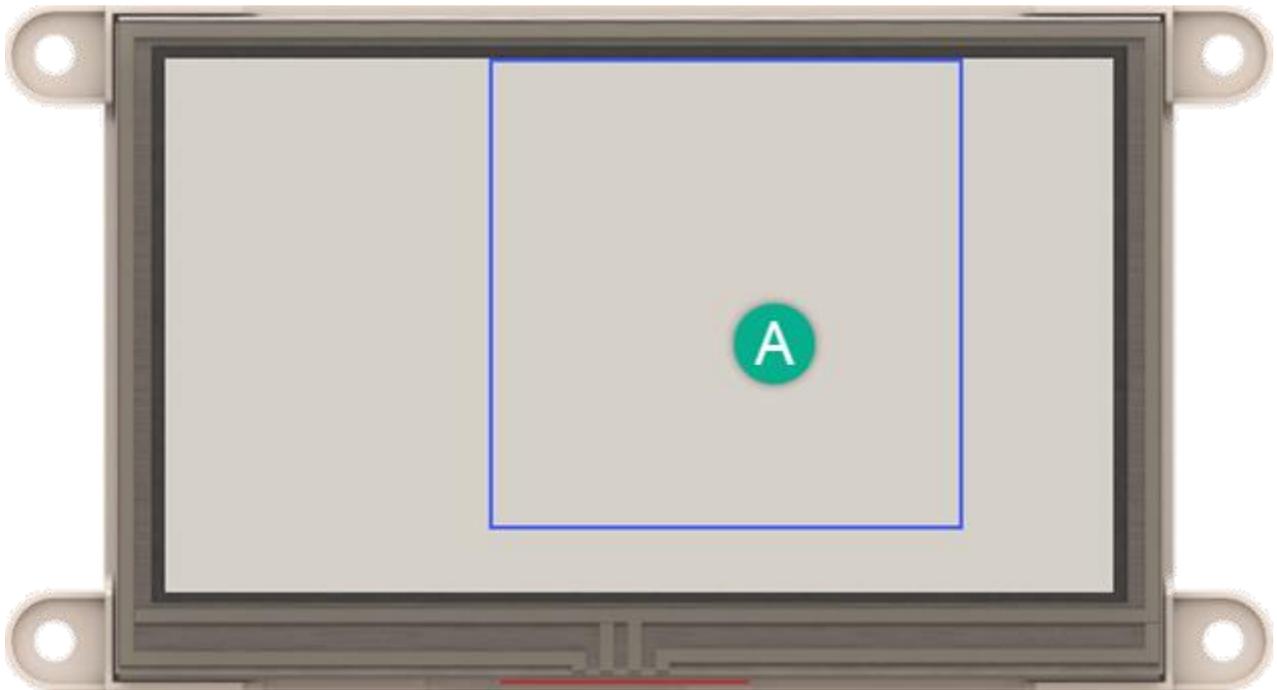
### 7.3.3. Change the Location of the Static Image

The x and y coordinates of the static images can be set by changing the values of the **Min/Max X and Min/Max Y** parameters.

Min/Max X	Centre	0.0	0.0
Min/Max Y	Centre	0.0	0.0

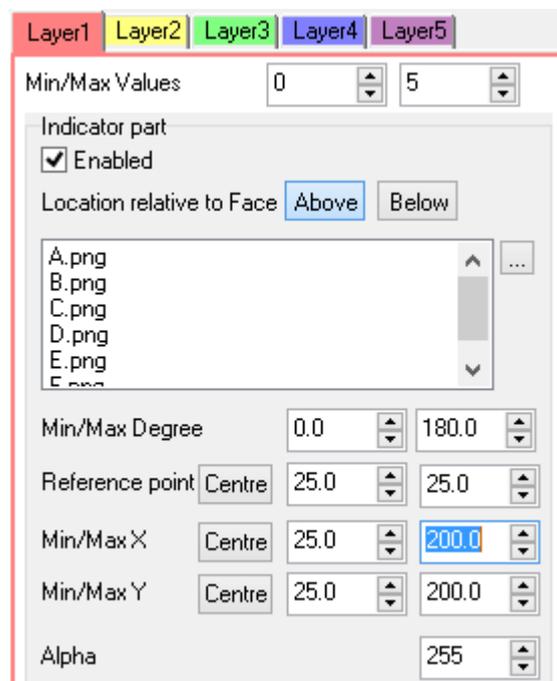
By default, a set of static images in a layer has **Min/Max X and Min/Max Y** parameters set to **0**. Thus, the top-left corners of the static images are located at point **(0,0)** of the working area. In the image below, the **Min/Max X and Min/Max Y** parameters are set to **(120, 120)** and **(120, 120)**.

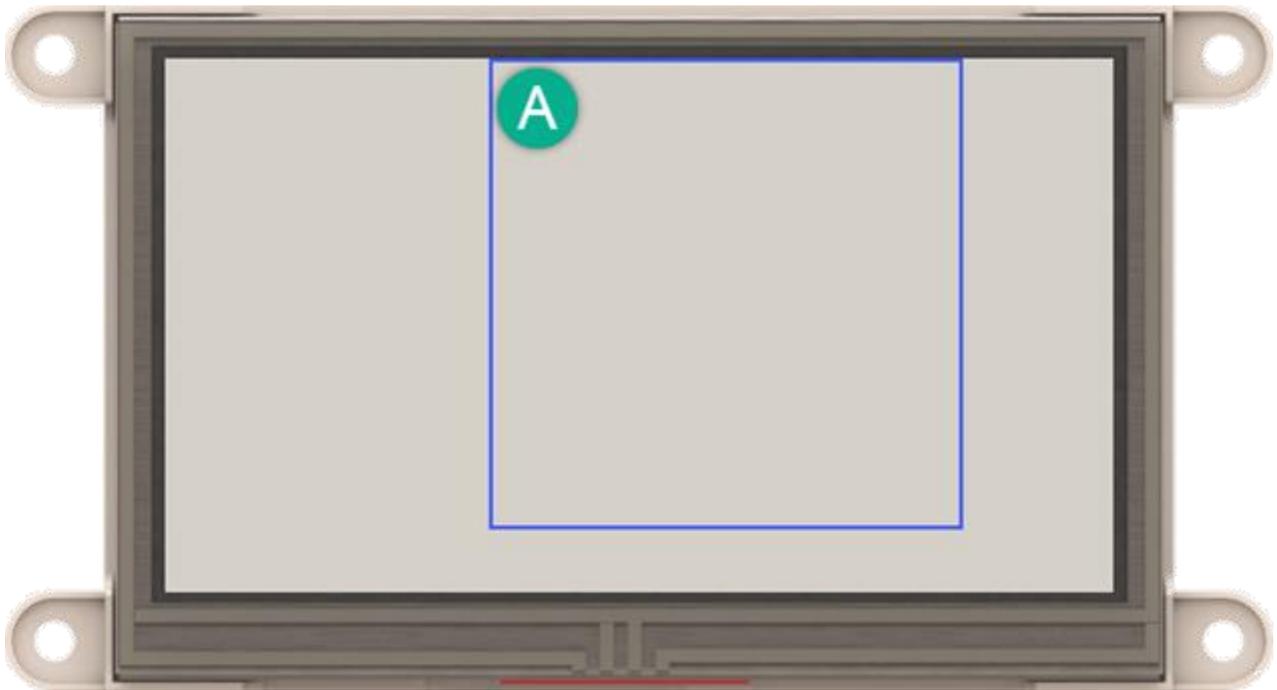
Indicator part			
<input checked="" type="checkbox"/> Enabled			
Location relative to Face		Above	Below
<ul style="list-style-type: none"> <li>A.png</li> <li>B.png</li> <li>C.png</li> <li>D.png</li> <li>E.png</li> <li>F.png</li> </ul>			
Min/Max Degree	Centre	0.0	0.0
Reference point	Centre	0.0	0.0
Min/Max X	Centre	120.0	120.0
Min/Max Y	Centre	120.0	120.0



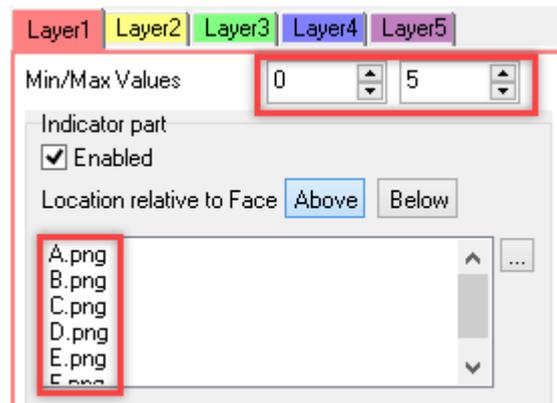
#### 7.4. Layers with a Series of Manipulated Images

It is also possible for a layer to contain a series of images that are manipulated at the same time. Consider the following example, which is similar to the previous example.





In this example, there are six images and six frames.

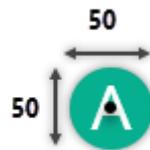


The **Min/Max Degree** parameters indicate that the images will be rotated from 0 to 180 degrees.



The **Reference point** parameters indicate that the images will be rotated at their center.





The **Min X and Y** and **Max X and Y** parameters indicate that the reference point of the images will be moved from **(25, 25)** to **(200, 200)**.

Indicator part

Enabled

Location relative to Face

A.png  
B.png  
C.png  
D.png  
E.png  
F.png

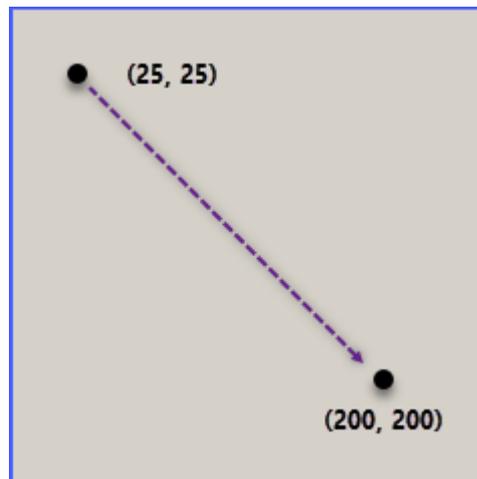
Min/Max Degree 0.0 180.0

Reference point Centre 25.0 25.0

Min/Max X Centre 25.0 200.0

Min/Max Y Centre 25.0 200.0

Alpha 255

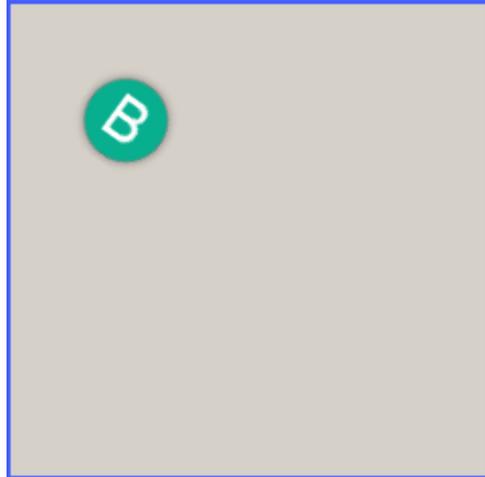


The above configuration will result to the following:

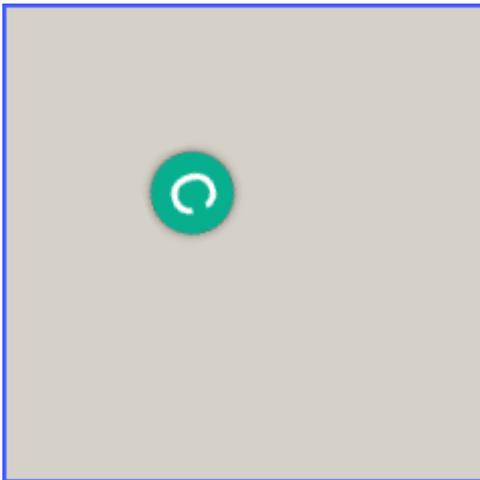
At frame 0:



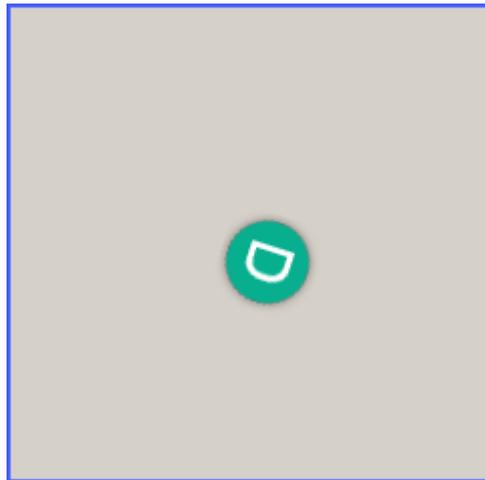
At frame 1:



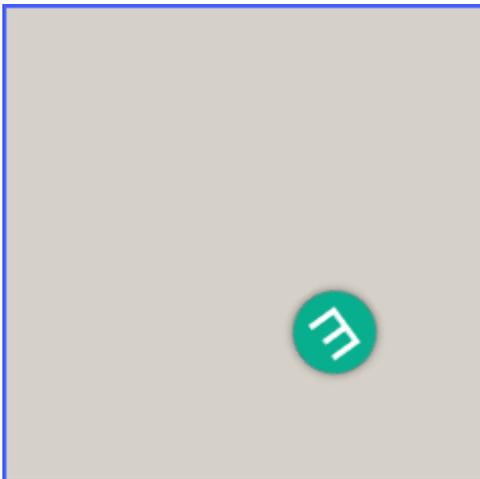
At frame 2:



At frame 3:



At frame 4:

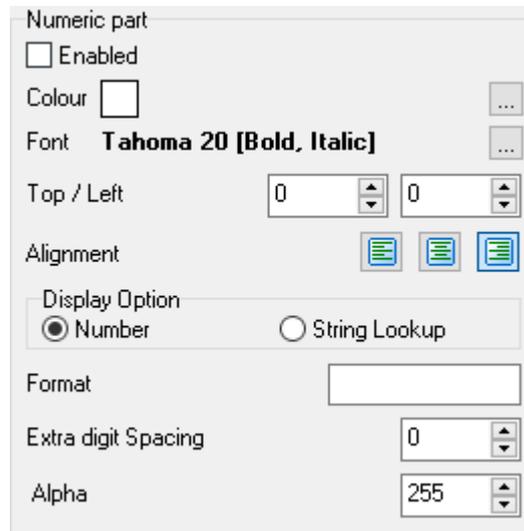


At frame 5:



## 7.5. Adding Numerical Part

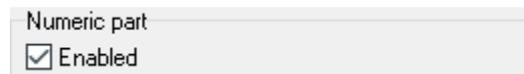
The value of each layer can be displayed on the widget using the **Numeric part** section.



The screenshot shows the 'Numeric part' configuration panel. It includes the following settings:

- Enabled
- Colour:  (with a color picker icon)
- Font: **Tahoma 20 [Bold, Italic]** (with a font picker icon)
- Top / Left: Two spinners, both set to 0.
- Alignment: Three icons for left, center, and right alignment.
- Display Option:  Number,  String Lookup
- Format:
- Extra digit Spacing:  0
- Alpha:  255

To display the value of a layer simply enable its numeric part.



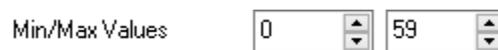
The screenshot shows the 'Numeric part' configuration panel with the 'Enabled' checkbox checked.

- Enabled

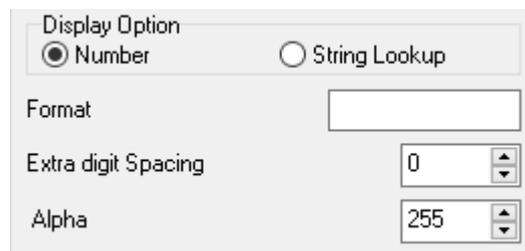
### 7.5.1. Display Options

There are two options for displaying the numeric part. These are **Number** and **String Lookup**.

**Number** corresponds to the minimum and maximum value of a layer (**Min/Max Values**). By using this, the value of which the layer is currently at on the GCI frame will simply be printed with its top left position at the **Top / Left** position on the working area.



The values printed can be formatted using **Format**.



By default (using "blank"), a value of zero displays nothing. Using *0* or *00* will display a minimum of 1 digit or 2 digits respectively. If a thousand operator is desired, a comma is simply needed to be in the **Format**.

**Format** also accepts decimal places. A **Format** of *0.0* will always display a number with a minimum of 2 digits with the decimal point just before the last digit. In that case, if the value of the layer is 5, the value printed will be 0.5

**Extra digit Spacing** provides the user an option to add extra pixel spacing between each printed digits.

**Alignment Options** are provided for the user. The user can easily left align, centre align or right align the numeric part with respect to the longest string to be generated.

**Alpha**, just like in the image part of each layer, controls the transparency. An alpha of 255 will give an opaque numerical part.

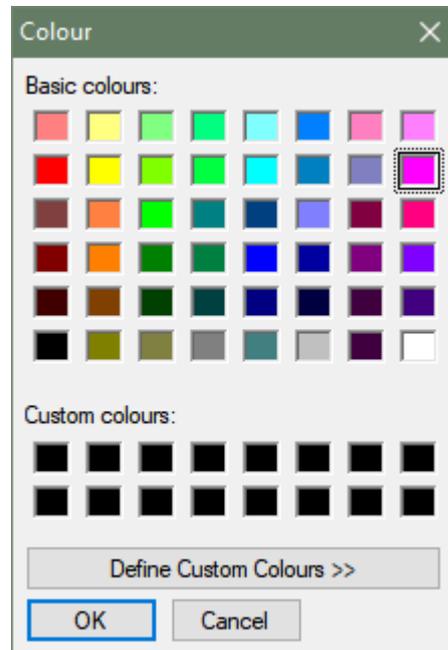
**String Lookup** corresponds to Strings that the values corresponds to. This is useful for widgets with small number of frames.



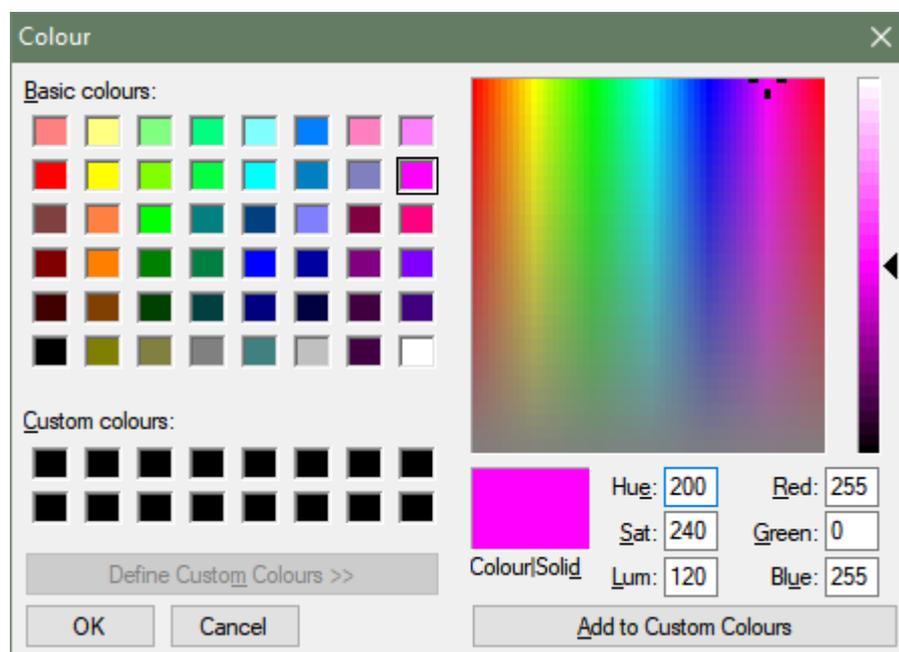
When using **String Lookup** with a manipulated image, previously set **Min/Max Values** will be ignored and be changed automatically depending on the number of **Strings** listed.

### 7.5.2. Changing the Font Colour

The font colour can be changed by clicking on  of **Colour** to open this window.

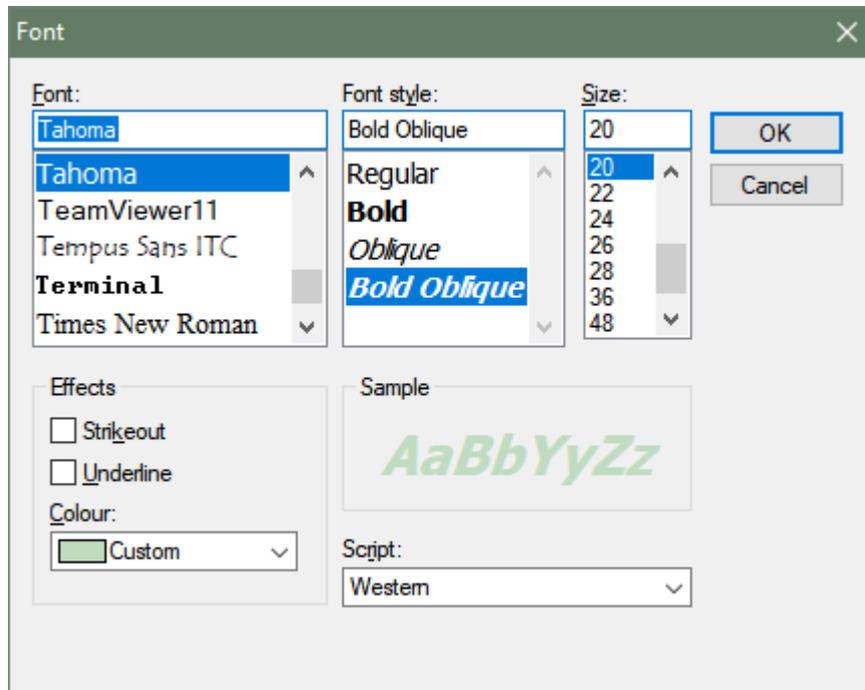


This allows users to select a basic colour or define a new Custom font colour. When defining a new custom colour, the window will open a colour picker



### 7.5.3. Changing the Font, Font Size and Font Style

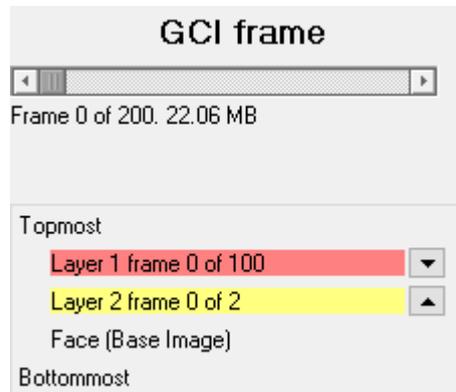
Font, font size and font style can be modified by clicking on  of **Colour**.



This window will open. Pressing **OK** will apply the selected font properties.

## 8. Linking Layers

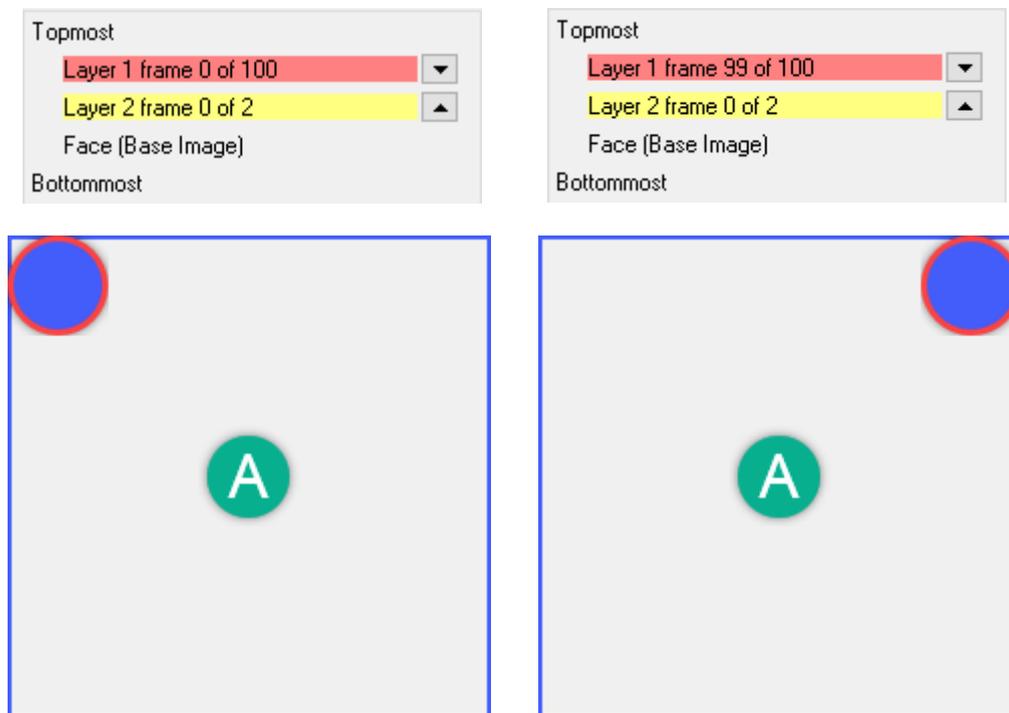
By default, frames are generated for all possible combinations of all the values of enabled layers. The layers can be linked if this isn't desired.



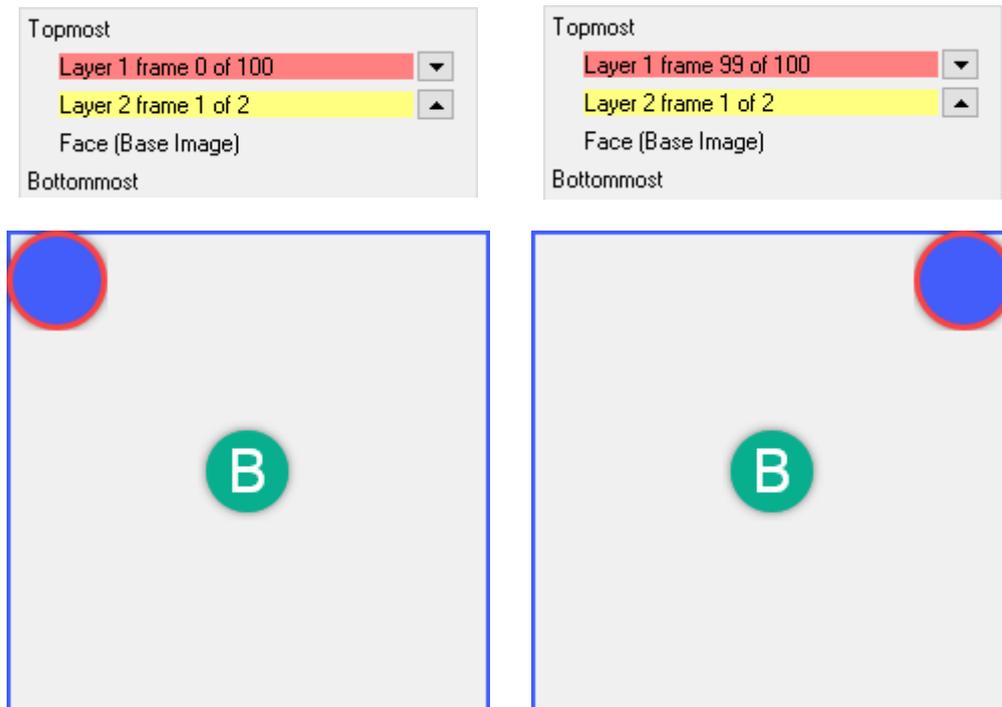
In this example, Layer 1 has 100 frames indexed 0 to 99 while Layer 2 has 2 frames indexed 0 and 1. Without linking the two layers, the GCI frame has an output of 200 frames indexed 0 to 199.

$$N_{Total} = N_1 \times N_2 \dots N_5$$

That is 100 frames from Layer 1 while Layer 2 is on its first frame (index 0).



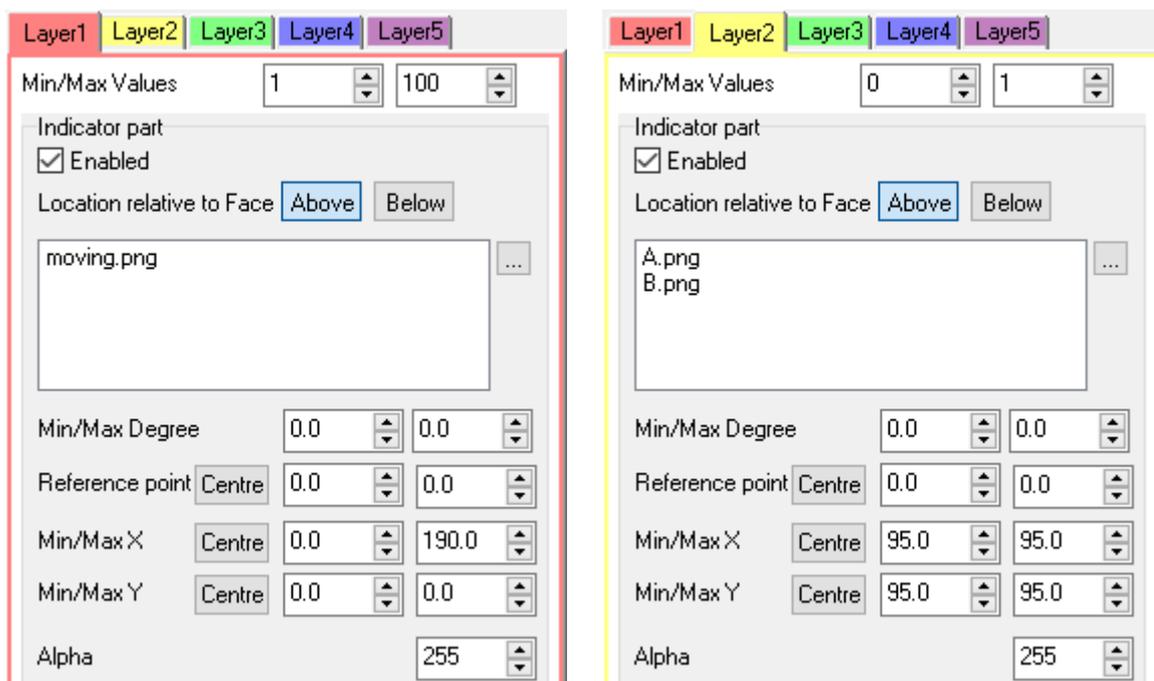
and another set of 100 frames from Layer 1 while Layer 2 is on its second frame (index 1).



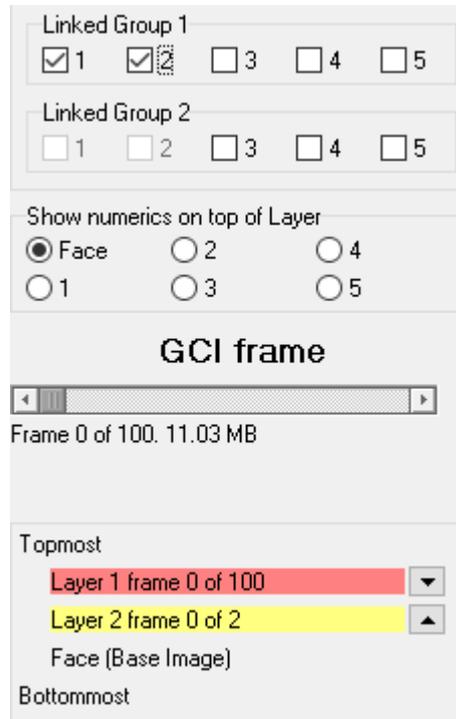
As shown in the images, it can be noticed that Layer 1 completes one cycle before Layer 2 moves to its next frame this. If other layers are active, this pattern is also applicable. This was discussed in [Order of Layers](#)

That is if Layer 1 completes one cycle then Layer 2 moves to its next frame, if Layer 2 completes one cycle then Layer 3 moves to its next frame, and so on. Single frame enabled layers can be simply skipped.

Here are the layers used on the previous example.



Linking layers allows layers to move at the same time with each other. Using the same example, linking the two layers will give a 100 GCI frames instead of 200.



It is possible to have up to two groups of linked layers as shown in the image.

Layer 1 contains a manipulated image having values 1 to 100 therefore having a total of 100 frames indexed 0 to 99 while Layer 2 contains two static images having default values 0 and 1.

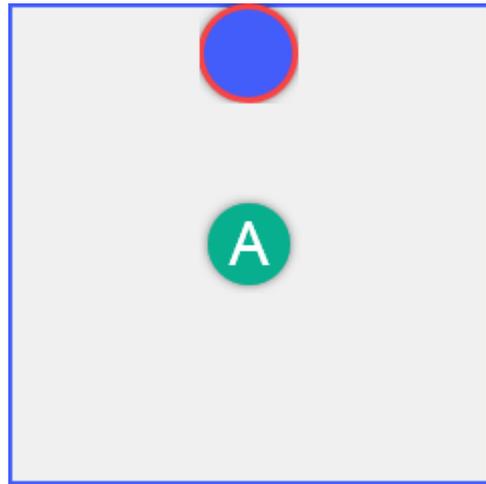
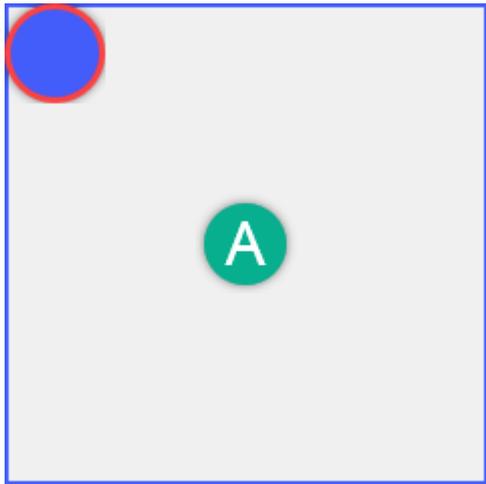
Instead of having two sets of 100 frames, by linking the layers, the output GCI will have the 100 frames from Layer 1 be equally divided among the 2 frames of Layer 2.

$$N_{Total} = G_1 \times G_2 \times N_a \dots N_b$$

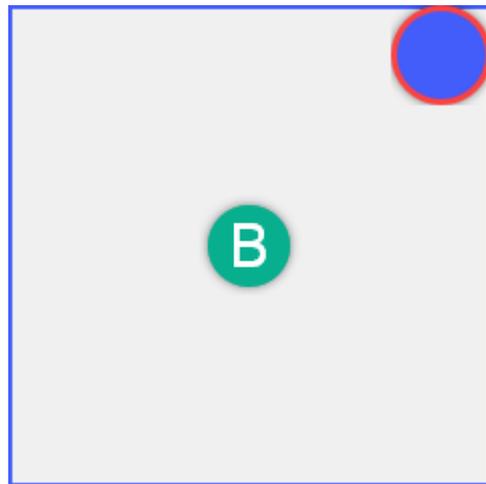
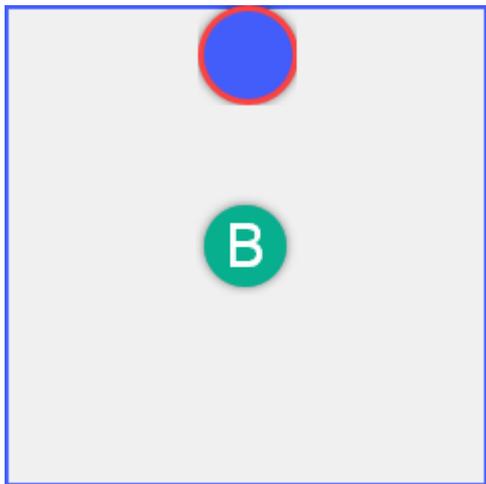
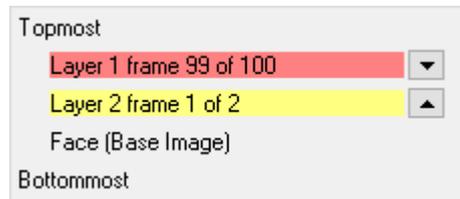
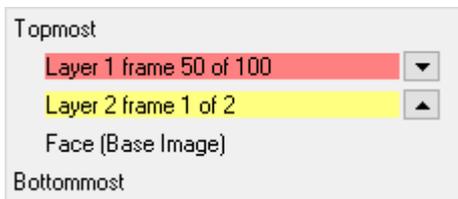
Where: G1 is the number of frames of the layer with the most frames from group 1 (if not used, G1 = 1)  
 G2 is the number of frames of the layer with the most frames from group 2 (if not used, G2 = 1)  
 Na to Nb are the frames not linked with any layer (if not used, N = 1)

Therefore, the first 50 frames (index 0-49) of Layer 1 will be generated with the first frame (index 0) of Layer 2





and the other 50 frames (index 50-99) of Layer 1 will be generated with the other frame (index 1) of Layer 2.



Order of linked layers with respect to other layers or other group of linked layers can simply be described as being the same as the order of the least indexed layer in the group.

For instance, a smart gauge with all layers enabled has Layer 1 and 5 linked together and Layer 3 and 4 linked together as well. The order of the layers will be:

Layer 1	Layer 2	Layer 3
Layer 5		Layer 4

Notice that Layer 5 simply joins with Layer 1 while layer 4 joins Layer 3. Or simply:

Group 1 (Layer 1)	Layer 2	Group 2 (Layer 3)
-------------------	---------	-------------------

This means Group 1 needs to complete a whole cycle from its **Min/Max Values** before Layer 2 moves a single frame. Then when Layer completes a whole cycle, Group 2 will move a single frame.

## 8.1. Clockwork Link

There is a special kind of layers link that is mainly for the purpose of creating analog clocks widgets using layers with manipulated layers though it is not limited to that. It can be used with other widgets as long as the user wants to.

Let's discuss this in detail using a Workshop4 example.

Open a Smart Gauge Tool. Load an existing smart widget file by clicking on  Load.

It will open this location

**\\Users\Public\Documents\4D Labs\Smart Gauges Knobs and Sliders**

Select and open the file **HMS Watch 281.4Dsmart**

This widget has three enabled frames for the clocks hands.

Check Layers 2 and 3 and you'll notice that both have this option checked

Clockwork link to previous layer

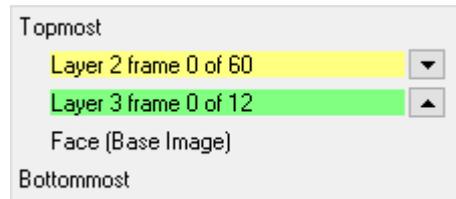
Before testing this out, observe the frames when this feature is not being used. Start by disabling the clockwork links.

Clockwork link to previous layer

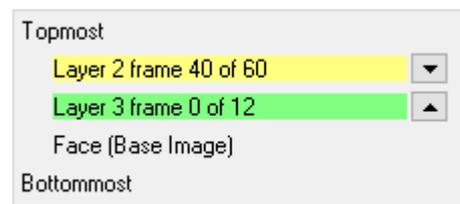
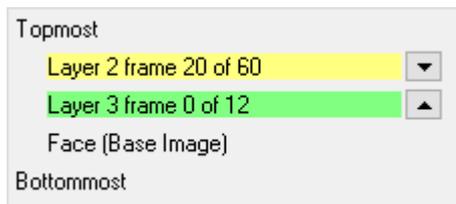
The effect is hardly noticeable for minute and second hand clockwork link. Therefore, let's disable Layer 1 (seconds' hand) for now and observe Layer 2 and Layer 3 for this example.



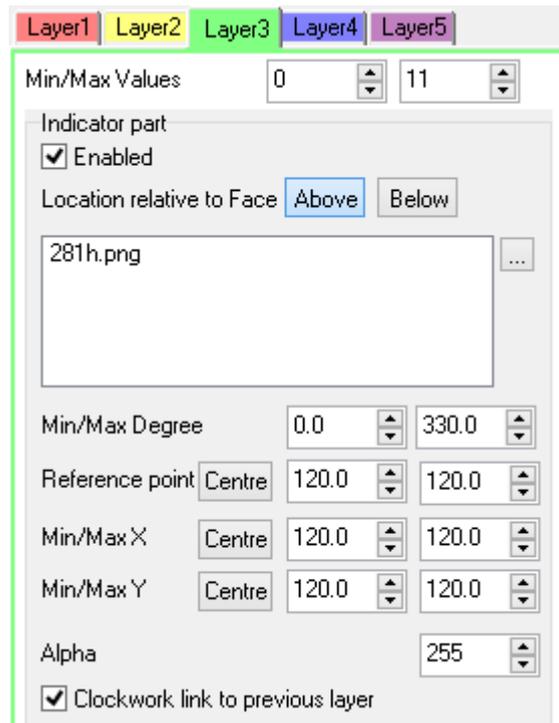
By doing that, now we have:



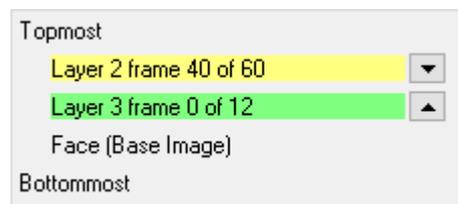
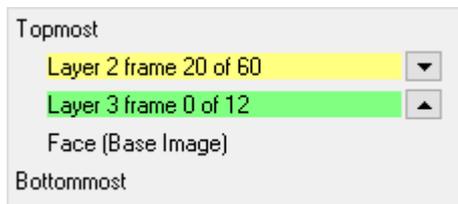
Similar to what's been discussed in **Linking Layers**, the first layer will complete a cycle first then the next layer will move one frame. In this case, once the minute hand (Layer 2) has completed a cycle, the hour hand (Layer 3) will move to the next frame which is the next hour.



However, as shown above, the output isn't desirable for a clock since the hour hand should rotate towards the 1-hour position while the minute hand is completing its cycle.



For that, users can use the Clockwork Link feature. Enable clockwork feature for Layer 3 then observe the difference.



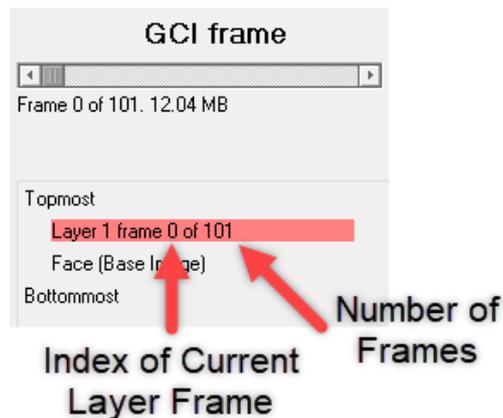
As shown above, the hour hand moves towards the 1-hour position while the minute hand is completing a whole 60 minute cycle.

## 9. Display Widget Frames

The frames in a layer are indexed from **0** to **N-1**, where **N** is the total number of frames. To display the frames, the GCI frame slider is used.



The layer guide shows all of the active layers and their order of arrangement. The index of the frame that is currently displayed and the total number of frames on each layer are also shown. In the example below, only Layer 1 is active. It has **100** frames and frame **0** is currently being shown.



Detailed discussions on the frames of a layer and how they are displayed are found in the sections [Layers with a Manipulated Image](#) and [Layers with Static Images](#).

Besides displaying frames of images, a layer can also display the corresponding values. For more information, refer to the section [Adding Numerical Part](#).

### 9.1. FAT16 File Size Limitation

The maximum file size limit for FAT16 is 2 GB. Hence, for widgets with a computed file size exceeding this limit, a warning message will be displayed under the GCI Frame slider.



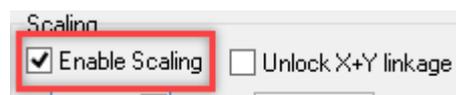
Reducing the number of layers and/or frames in a layer will reduce the resulting file size. For fully working analog clock widgets however, the minimum total number of frames is 43200 (60 x 60 x 12). To be able to use a fully working analog clock widget, users can reduce the dimensions of the clock to around 150x150 pixels. This will limit the resulting file size to around 2 GB and below.

## 9.2. Scaling Widgets

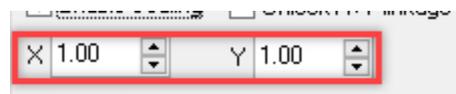
The scaling section at the bottom part of the Smart Widgets Editor window provides a facility to upscale or downscale the widget.



However, for best quality and speed of generation, the user is advised to start with source images that are of the proper size already. To scale a widget, enable scaling as shown below.

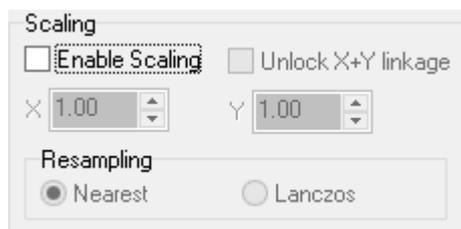


Input the X and Y scaling parameters.

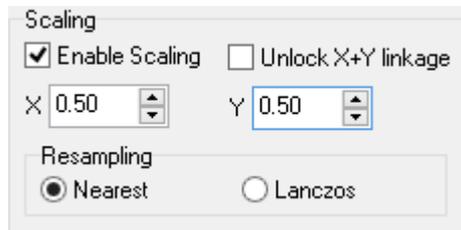


These parameters are the multiplying factors of the X and Y dimensions of the widget. Thus, a value of "1.00" corresponds to "100%" of the dimension. A value of "0.50" corresponds to "50%" of the dimension. The following images show a widget with three different scaling parameters.

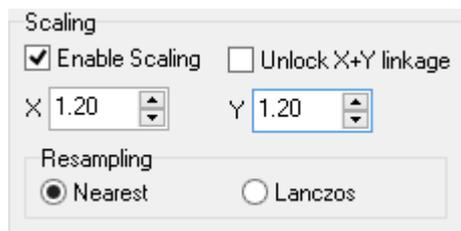
**Widget is not scaled:**



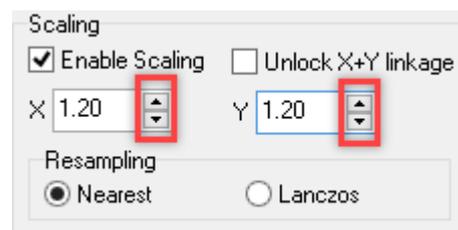
Widget is 50% of its original size:



Widget is 120% of its original size:



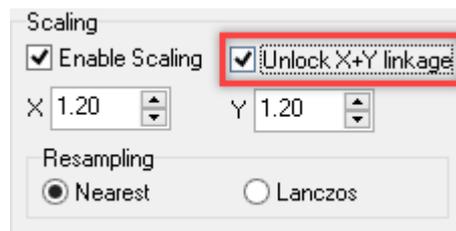
When adjusting the scaling parameters, the user can click on the up/down arrows beside the parameter values.



It is also possible to use the up and down arrows of the keyboard or the scroll wheel of the mouse. All of these options adjust the scaling parameters by **0.01**. On the other hand, using the PgUp and PgDn buttons of the keyboard adjusts the scaling parameters by **0.10**.

### 9.2.1. Disable Proportional Scaling

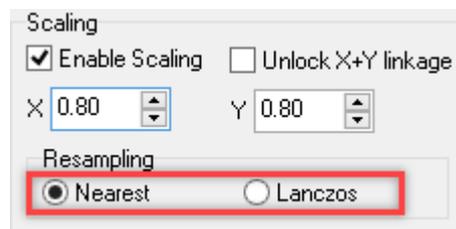
By default, the scaling of a widget is proportional, such that any change in the X scaling parameter is automatically reflected in the Y scaling parameter. To disable this, click on the tick box beside **Unlock X+Y linkage** property.



The X and Y parameters can now be adjusted independently.

### 9.2.2. Resampling Options

There are two resampling options – **Nearest** and **Lanczos**.



**Nearest** is the default option. It is low quality, and it will not significantly affect the build time. On the other hand, **Lanczos** resampling is high quality, and it will significantly extend build time. Again, for best quality and fastest speed of generation, the user is advised to start with source images that are of the proper size already.

## 10. Exploring Possibilities

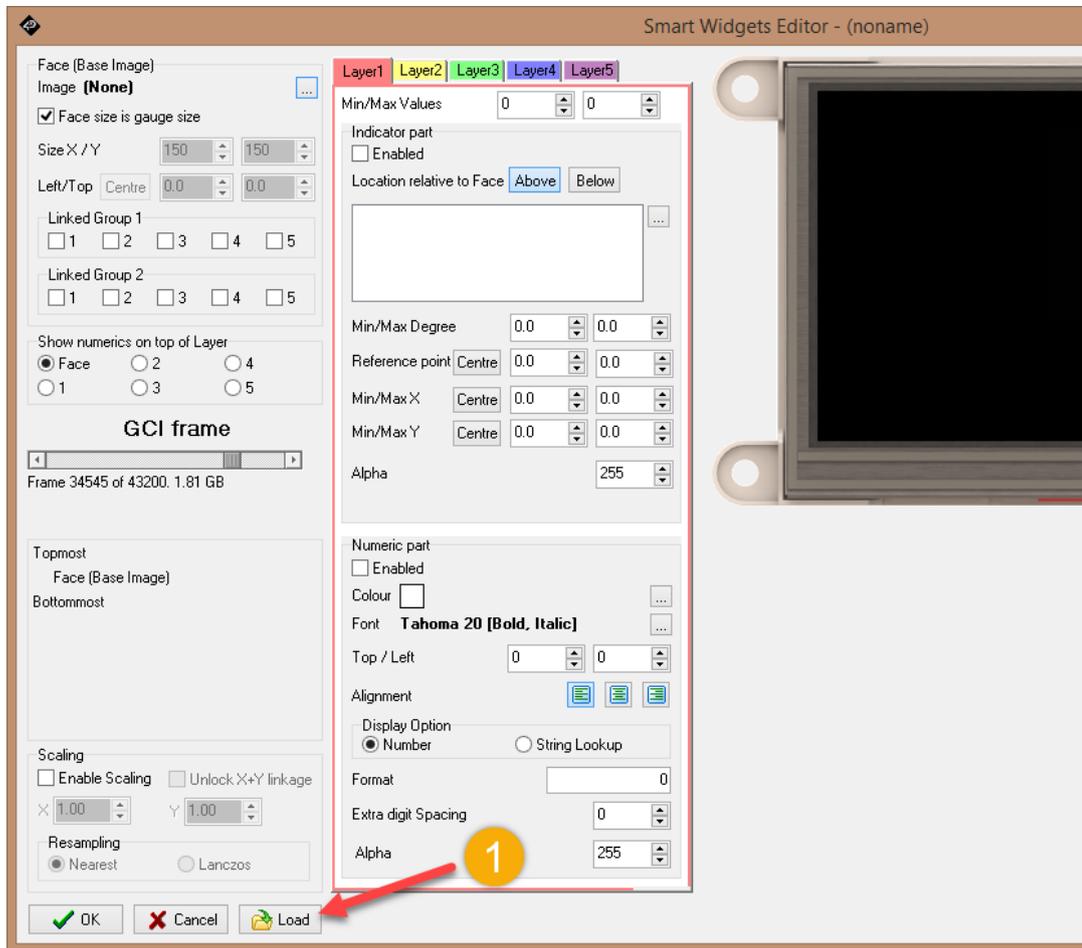
The Smart Widgets Editor tool is a powerful utility which enables the generation of complex custom widgets. Generally, the process of creating such widgets can be broken down in to the following steps:

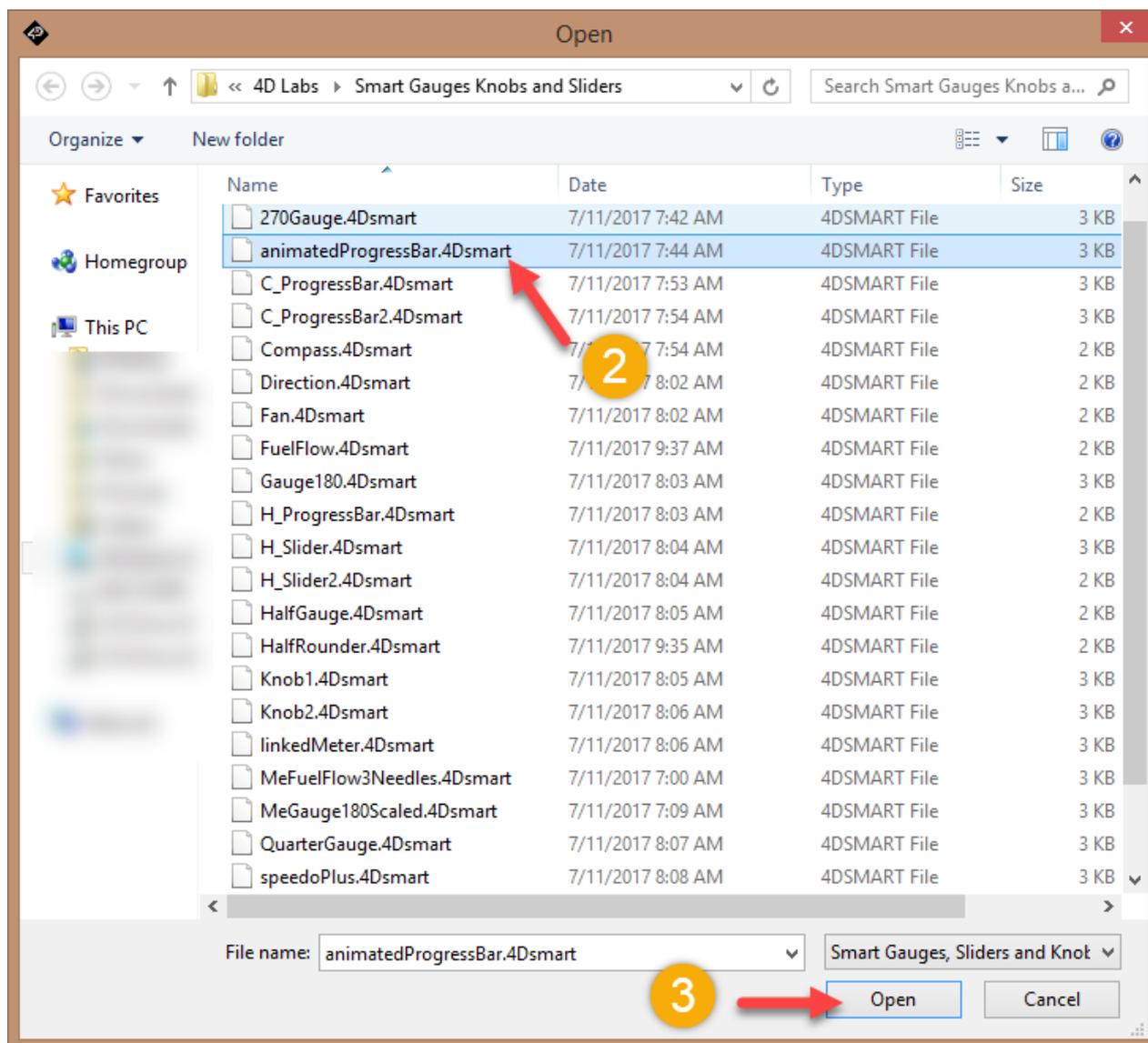
1. Determine the number of layers needed for the widget. Generally a moving part needs a separate layer.
2. Determine the different motions needed to animate each layer (horizontal, vertical, angular, or a combination of any of these).
3. Create the graphics for each layer using an external tool such as Paint, Gimp, or Photoshop. Add the image to the corresponding layer.
4. Implement and test the animation of each layer by playing with the manipulation parameter values.
5. Link layers to each other if necessary.

In some cases, the number of layers can be reduced further by a careful analysis of how the widget is animated. For instance, layers that have the same motion could be combined. Clever use of transparency and static images can also be utilized to reduce the number of layers.

10.1. Workshop4 Examples

Workshop4 comes with plenty of examples of Smart Widgets. To see the examples, follow the procedure below.





## 10.2. Application Notes

Application notes provide step-by-step discussions on how to create basic Smart Widgets. Click on any of the links below to see an application note.

[Smart Widgets: Horizontal Slider](#)

[Smart Widgets: Vertical Slider](#)

[Smart Widgets: Horizontal Progress Bar](#)

[Smart Widgets: Circular Progress Bar](#)

[Smart Widgets: Knob](#)

[Smart Widgets: 300-degree Gauge](#)

[Smart Widgets: Quarter Gauge](#)

[Smart Widgets: Half Rounder Gauge](#)

[Smart Widgets: Compass](#)

[Smart Widgets: Tank](#)

## 11. Revision History

Revision	Revision Content	Revision Date
1.0	First release, 4D Systems	July 25, 2017
S 1.1	Updated information on numeric part and order of layers	July 29, 2017

## 12. Legal Notice

### Proprietary Information

The information contained in this document is the property of 4D Labs Semiconductors and may be the subject of patents pending or granted, and must not be copied or disclosed without prior written permission.

4D Labs Semiconductors endeavours to ensure that the information in this document is correct and fairly stated but does not accept liability for any error or omission. The development of 4D Labs Semiconductors products and services is continuous and published information may not be up to date. It is important to check the current position with 4D Labs Semiconductors. 4D Labs Semiconductors reserves the right to modify, update or makes changes to Specifications or written material without prior notice at any time.

All trademarks belong to their respective owners and are recognised and acknowledged.

### Disclaimer of Warranties & Limitation of Liability

4D Labs Semiconductors makes no warranty, either expressed or implied with respect to any product, and specifically disclaims all other warranties, including, without limitation, warranties for merchantability, non-infringement and fitness for any particular purpose.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications.

Images and graphics used throughout this document are for illustrative purposes only. All images and graphics used are possible to be displayed on the 4D Labs Semiconductors range of products, however the quality may vary.

In no event shall 4D Labs Semiconductors be liable to the buyer or to any third party for any indirect, incidental, special, consequential, punitive or exemplary damages (including without limitation lost profits, lost savings, or loss of business opportunity) arising out of or relating to any product or service provided or to be provided by 4D Labs Semiconductors, or the use or inability to use the same, even if 4D Labs Semiconductors has been advised of the possibility of such damages.

4D Labs Semiconductors products are not fault tolerant nor designed, manufactured or intended for use or resale as on line control equipment in hazardous environments requiring fail – safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines or weapons systems in which the failure of the product could lead directly to death, personal injury or severe physical or environmental damage ('High Risk Activities'). 4D Labs Semiconductors and its suppliers specifically disclaim any expressed or implied warranty of fitness for High Risk Activities.

Use of 4D Labs Semiconductors' products and devices in 'High Risk Activities' and in any other application is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless 4D Labs Semiconductors from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any 4D Labs Semiconductors intellectual property rights.

## 13. Contact Information

For Technical Support: [www.4dsystems.com.au/contact](http://www.4dsystems.com.au/contact)  
For Sales Support: [sales@4dsystems.com.au](mailto:sales@4dsystems.com.au)  
Website: [www.4dsystems.com.au](http://www.4dsystems.com.au)

Copyright 4D Labs Semiconductors 2000-2017.