

MOPS

Mops specify the mask operations to be performed on the empty space skipping volume. This volume is used as a mask. The rendering is limited to within the non-zero regions of this mask. This buffer is generated by applying contributing transfer functions to *drag* volume. This buffer has 3 channels 0,1 and 2.

Channel 0 has 255 for those voxels that contribute to the final image and 0 for the rest. Channel 1 stores the opacity information.

Channel 2 is used to for painting.

In the final rendering, values in channel 0 from mask buffer are multiplied with the opacities. 0 value means completely transparent - will not contribute to final image. For non-zero values in channel 2 (which stores labels/tags), appropriate label/tag color and opacity (change label or tag color/opacity via Preferences widget) is mixed to generate the final color and opacity.

The mask buffer is saved in keyframes (when there is a mop operation performed on the mask buffer), so they can be animated. The mask buffer resides on GPU and needs to be downloaded onto CPU side. GPU-CPU transfers are very slow, therefore saving and restoring keyframes will be slower for large datasets when mop is in use. When there is no mop involved the mask buffer is not saved - default case. Also, user has option of not interpolating the mask buffer - use "None" option for "Interpolate Mop" available when spacebar is pressed for selected keyframe. Mask buffers can also be copied from one keyframe to another.

Morphological operations can be performed in stages. For example to remove top skin one can follow the following protocol :

shrinkwrap (mop shrinkwrap 10) based on size of the hole you would like to plug.

invert (mop invert) invert current mask

thicken (mop thicken 4) thicken current mask - this will catch 4 voxel deep top layer

invert (mop invert) this will expose interior by removing top layer - that thicken 4 voxel layer.

Another way could be

shrinkwrap (mop shrinkwrap 10)

cityblock (cityblock 4) distance transform will be evaluated only 4 voxels deep

saturate 0 (mop sat0 4) voxel masks below 5 are set to 0 - this will remove 4 voxel deep top layer.

Users can also specify transfer function set to use while creating the mask buffer.

Mop operations can be given in two ways via main Command Dialog panel -

1. mop (this opens up command panel for mops)
2. mop [option] [parameters] (this will execute the operation without opening the mop command panel).

Various options available under mop –

average	average [chan1] [chan2] [dst] Save the average value of two channels. $dst = (chan1 + chan2)/2$
blend	blend [off] switch on/off blend mode.

	<p>When blend mode is switched on, the paint tags that users have painted on the data are used as reference to the transfer functions for blending. Appropriate transfer functions will be blended in place of tag colors - tag value refers to the transfer function set. If the tag value is greater than the total number of transfer function sets, then set 0 is chosen.</p> <p>When multiple volumes are loaded, appropriate transfer functions from different sets are applied to the volumes with the tag value referring volume 0 transfer functions. In the case of multiple volumes, user need to take care of choosing appropriate tag values for painting, so that there is no overlap between transfer function sets.</p>
carve	<p>carve [off] switch on/off carve mode.</p> <p>Using carve users can modify values in channel 1. Only the mask values in a spherical region around mouse cursor and those that have non-zero channel 1 are affected. Channel 1 values are normally set for non-zero opacity voxels when mask texture gets updated with transfer function change.</p> <p>When in carve mode, users can remove spherical portions of the data by using shift+left mouse drag. Normally shift+left mouse click add a point - this normal function is disabled when carving is switched on. Only mask values that have non-zero channel 1 are carved.</p> <p>Shift+right (or Ctrl+left) mouse drag will restore mask buffer values in the spherical region defined by carve radius. Only mask values that have non-zero channel 1 are restored.</p> <p>Shift+middle (or Alt+left) mouse drag will set mask buffer values in the spherical region defined by carve radius to 255.</p> <p>Users can also employ paths and points for carving operations. Path can also be used to patchup big holes. Points can be used to fillup large empty spaces before shrinkwrap operations.</p>
carverad	<p>carverad [radius] [decay]</p> <p>Specify carve radius and decay parameters. Carving allows users to remove spherical portions of the data by setting mask buffer values to 0 in that region. Carve radius defines the sphere of influence. The decay parameter specifies the smoothing near the edges - mask buffer values are set to 0 near the central region and are smoothstepped to original values near edge. Decay value 0 implies sharp boundary.</p> <p>Users can also change carve radius and decay using up/down and left/right arrows respectively.</p> <p>For example mop carverad 20 5 : influence sphere of radius 20, with 0 values within sphere of radius 15 and gradually climbing to original values near the edge.</p>

chessboard	<p>chessboard [sz]</p> <p>Chessboard distance transform. The values start with 0 on the surface of the object and increasing inside the object. Default value for sz is 1.</p>
cityblock	<p>cityblock [sz]</p> <p>Cityblock distance transform. The values start with 0 on the surface of the object and increasing inside the object. Default value for sz is 1.</p>
close	<p>close [sz] [chan]</p> <p>Morphological closing of channel chan - dilate followed by erode dictated by size parameter. Default value for sz is 1 and chan is 0.</p>
copy	<p>copy [src] [dst]</p> <p>Copy mask buffer channel src to mask buffer channel dst. src and dst can be either 0,1,2.</p>
copyfromsaved	<p>copyfromsaved [src] [dst]</p> <p>Copy savedbuffer channel src to mask buffer channel dst. If specified, src and dst can be either 0,1,2. If not specified, all channels from savedbuffer are copied to mask buffer.</p>
copytosaved	<p>copytosaved [src] [dst]</p> <p>Copy mask buffer channel src to savedbuffer channel dst. If specified, src and dst can be either 0,1,2. If not specified, all channels from mask buffer are copied to savedbuffer.</p>
dilate	<p>dilate [sz] [chan]</p> <p>Dilate mask buffer channel chan by sz. Default is dilate by 1 for chan 0.</p>
dilateedge	<p>dilateedge [val1] [val2]</p> <p>Very similar to erode - values near the edge are set to zero based on current mask values at those positions. User needs to provide two parameters to this operation which will specify range of values that can be eroded. If only one value is provided then the range of values is taken as [1, val].</p> <p>For example mop dilateedge 100 : erode edge where values are between 1 and 100 mop dilateedge 230 253 : erode edge where values are between 230 and 253</p>

edge	<p>edge [val] [thickness]</p> <p>Set edges of mask buffer to val. Parameter thickness controls how thick edge would be. Default val is 0 and thickness is 1.</p> <p>For example mop edge 0 2 : set 2-voxel thick edge to 0</p>
erode	<p>erode [sz] [chan]</p> <p>Erode mask buffer channel chan by sz. Default value for sz is 1 and chan is 0.</p>
hatch	<p>hatch [box grid] [xn] [xd] [yn] [yd] [zn] [zd]</p> <p>This action etched a box or grid hatching pattern on channel 0 based on "box" or "grid" option and integer values xn,xd,yn,yd,zn,zd.</p> <p>Pseudo code to set channel value to 0 is as follows :</p> <pre> zeroit = true; if (xn > 0 and xd > 0) zeroit .and. (xcoord%xn < xd); if (yn > 0 and yd > 0) zeroit .and. (ycoord%yn < yd); if (zn > 0 and zd > 0) zeroit .and. (zcoord%zn < zd); if (box selected and zeroit is true) then set value to 0; if (grid selected and zeroit is false) then set value to 0; </pre>
histogram	<p>histogram [chan]</p> <p>Calculate frequency histogram for channel chan. Default value for chan is 2.</p>
fusepatch	<p>merge patches that were added using paths with the rest of the mask. Patches are usually added to help contain the interior region during shrinkwrap process.</p>
invert	<p>Inverts the mask buffer : 255-current mask value.</p>
localmax	<p>Find locally maximum values.</p>
localthickness	<p>localthickness [sz]</p> <p>Finds local thickness - channel 0 mask buffer values are used to calculate the local thickness and these values are stored in channel 2 of the mask buffer. Default value for sz is 100.</p>
masktf	<p>masktf [tf]</p> <p>Specify which transfer function set to use for generating the mask buffer. If the "tf" parameter is -1, then contributing transfer functions from all the sets are considered for generating the empty space skipping mask buffer. -1 is default value, if no value is specified.</p>

	<p>For example :</p> <p>mop masktf -1 - use transfer function from all the sets.</p> <p>mop masktf 1 - use transfer funtions from set 1 only.</p>
max	<p>max [ch1] [ch2]</p> <p>Take maximum of current mask buffer channel ch1 and saved buffer channel ch2 - behaves as union of current buffer and saved buffer. If specified, ch1 and ch2 can be either 0,1,2. If not specified, all channels are considered.</p>
maxvalue	Displays maximum buffer value for each channel.
min	<p>min [ch1] [ch2]</p> <p>Take minimum of current mask buffer channel ch1 and saved buffer channel ch2 - behaves as intersection of current buffer and saved buffer. If specified, ch1 and ch2 can be either 0,1,2. If not specified, all channels are considered.</p>
nop	Dilate mask buffer by 1. This is the default option.
open	<p>open [sz] [chan]</p> <p>Morphological opening of channel chan - erode followed by dilate dictated by size parameter.</p> <p>Default value for sz is 1 and chan is 0.</p>
paint	<p>paint [off]</p> <p>switch on/off paint mode.</p> <p>When in paint mode, users can paint tags on the data. Tags are shown as colours mixed on top of the transfer function. Tag colours are mixed based on opacity parameter of the colour, which can be set individually for each tag color. The tag colours are set via Preferences - Tag Colour menu.</p>
rdilate	<p>rdilate [val] [sz]</p> <p>Restricted dilate - dilate mask buffer by "sz" only for voxels with value "val".</p> <p>Default values for sz and val are 1 and 255 resp.</p>
removepatch	<p>Remove the patches that were added using paths.</p> <p>Patches are usually added to help contain the interior region during shrinkwrap process.</p>
sat0	<p>sat0 [val] [chan]</p> <p>Mask value of channel chan less than val will be set to 0. Some mask value will be greater than 0 but less than 255 for cityblock, thicken or shrinkwrap operations.</p> <p>Default value for val is 255 and chan is 0</p>

sat1	<p>sat1 [val] [chan]</p> <p>Mask value of channel chan greater than val will be set to 255. Some mask value will be greater than 0 but less than 255 for cityblock, thicken or shrinkwrap operations.</p> <p>Default value for val is 0 and chan is also 0.</p>
save	<p>Save mask buffer to raw file or image. The data will be save as 1-byte per voxel. The mask value will range from 0 to 255.</p>
itk	<p>Apply ITK (Insight Tool Kit) filters.</p> <p>Several filter are implemented. Channel 1 (which usually stores opacity information) is passed on to the filter. The results are passed back in channel 2. The filter is performed over the opacity information - users can tweek data in channel 1 (for e.g. carving) to achieve the required result.</p>
setvalue	<p>setvalue [val] [minval] [maxval] [chan]</p> <p>If channel chan maskbuffer value lies between minval and maxval, then set it to val.</p> <p>Default value for chan is 0.</p> <p>This option can be used to create a shell - for e.g. apply distance transform : mop cityblock 20 set voxels away from surface to 0 : mop setvalue 0 5 255 - all 5 and above voxel deep are set to 0 set non-zero voxels to 1 : mop sat1 0 - (or : mop setvalue 255 0 255)</p>
shrink	<p>shrink [sz] [chan]</p> <p>Erode channel 0 by sz based on values in channel chan. Erosion operation is only applied at voxels where value of chan is zero. Default value for sz is 10 and chan is 1.</p> <p>This operation is useful for eroding filled cavities on the exterior of the structure by shrinkwrap operation.</p>
shrinkwrap	<p>shrinkwrap [sz]</p> <p>Very similar to close, except erosion starts from the edge of the subvolume box. Default value for sz is 1. This operation applies following protocol :</p> <p>thicken current buffer by sz set 1-voxel thin edges of the bounding box to 0 thicken 0 valued edge by sz</p>
smoothchannel	<p>smoothchannel [chan]</p> <p>Apply 3x3x3 averaging for mask buffer values in channel "chan". Only non-zero values are considered for smoothing.</p>

swap	swap [ch1] [ch2] Swap mask buffer channel ch1 and savedbuffer channel ch2.
tag	tag [val] Set tag value for painting. Value must be between 0 and 255.
thicken	thicken [sz] Very similar to dilate - except mask buffer values decrease as one goes into empty region away from original surface like distance transform. Default value for sz is 1.
update	update [off] Switch on/off updating of mask buffer. By default the mask buffer will be updated every time a transfer function is changed. Users can switch off these updates and freeze the mask buffer by specifying "off" or "no" for the parameter. If no parameter is specified the updates will be switched on. The mask buffer will always be updated whenever any morphological operation (listed below) are performed. For example : mop update off - to turn off updates to mask buffer. mop update - to turn on updates for every transfer function change. Even when updates to the buffer via transfer function changes are switched off, all morphological operations do update the current buffer. This enables user to perform multiple operations one after another.
xor	xor [ch1] [ch2] XOR mask buffer channel ch1 with savedbuffer channel ch2. Result is stored in channel ch1 of mask buffer. If specified, ch1 and ch2 can be either 0,1,2. If not specified, all channels from mask buffer are xored with savedbuffer.