Lis of example widgets that can be generated with Omega

Type	Prompts:	Dataset or Image to use:	Notes:
Filtering	Please make me a Gaussian blur widget with sigma parameter	E.g., the 'Human Mitosis' built-in napari sample dataset.	
	Please make me a widget for spatially variable Gaussian blur with a sigma parameter varying over z between two values: sigma_bottom and m sigma_top. The top corresponds to high-z values, and the bottom corresponds to low-z values. Sigma values should range between 0 (no blur) and 10.	3D image	
	Please make a widget that applies Butterworth filtering to a 2D grayscale image with a defined cut-off frequency parameter normalized between 0 (min. freq.) and 1 (max. freq.) and order (between 1 and 10).	2D grayscale image	
	Please make a Sobel edge filter widget that works on 3D grayscale images. The user can choose an isotropic kernel size of 3, 5, or 7. Make sure to optimize the code.	3D image	
	Create a sharpening filter widget for 2D monochrome images and expose relevant parameters. Ensure that the image's total brightness is preserved. Expose a parameter to control the strength of the sharpening.	2D image	
	Make a band-pass filter widget that keeps frequencies in a 2D or 3D grayscale image between two frequencies: f_min, f_max. The frequencies are provided in the normalized range [0f, 1f].		
Transforms	Please make a widget that can crop a 3D image by specifying the number of pixels to remove on all sides (x-, x+, y-, y+, z-, z+).	3D image	
	Please make a widget that can up- and down-scale a 3D image anisotropically along x, y, z. The type of interpolation for each axis can be set independently.	3D image	
	Please make a widget that takes a 2D RGB image and returns one of the following layers: either one for the hue, for the saturation, or luminance (user choice). The resulting layer values should range between 0f and 1f for luminance and saturation but be RGB (fully saturated, maximal luminance) for the hue.	2D RGB image	
	Please write a widget to convert a 2D image from RGB to grayscale. The weights used to project the color pixels to grayscale are configurable. Ensure weights sum to 1. Add a gamma parameter for each component R, G, and B before projection. Apply the inverse geometric average gamma transformation to the resulting gray-scale value.	2D RGB image	
	Please make a widget that computes the max projections of a 3D monochrome image along an arbitrary axis parametrized by Euler angles. Use the Fourier transform and the projection-slice theorem. Pad the input image to have the same dimension along all axes if necessary.	3D image	
	Make a widget that fuses two 2D images using the wavelet transform. Offer the choice of transform. Expose relevant parameters.	Two 2D grayscale images.	You will need to install PyWavelets and restart napari for Omega to be able to find it. Omega will attempt to install.

Type	Prompts:	Dataset or Image to use:	Notes:
		2	
Enhancement	Please make a widget that applies CLAHE to a 2D image. Expose all relevant parameters.	2D grayscale image	
	Please write a geometric mean filter widget for denoising an image with an arbitrary number of dimensions.	2D grayscale image	
	Please write an alpha-trimmed mean filter to denoise a 2D image and expose relevant parameters if needed.	2D grayscale image	
Analysis	Please make a widget that, given a 2D+t image, computes the 2D+t RGB image in which the hue is proportional to the direction of the displacement as computed by PIV, and the luminance is proportional to the amount of displacement	2D+t image with clear flow.	
	Please make a widget that segments a 2D RGB image using the SLIC superpixel algorithm. Expose the number of segments as a parameter.	2D RGB image	
	Please make a widget that can apply hole filling to an arbitrary nD label image.	Labels layer	
	Please make a widget that detects lines in a 2D gray-level image.	2D grayscale image	
	Please make a widget that computes the binary skeleton of a 2D image.	2D grayscale image	
	Please make a widget that takes two 2D images, where the second is typically smaller than the first, and finds occurrences of the second image by correlation. The widget should return the first image but masking all pixels that are not covered by an occurrence of the second image.	Two 2D grayscale image, one being a crop other the other (for testing).	
	Please write a widget approximating a 2D gray-level image as the sum of gray-level Gaussians. Each Gaussian has a position, sigma, and intensity value that must be optimized to match the image best. Return the best-approximated image. The number of optimization steps (<1000), Gaussians (<1000), and the method to compare the approximated and input image should be parameters (MSE, etc…).	2D grayscale image	
	Please make a widget that segments an image using Otsu thresholding and returns a labels layer. If appropriate, expose relevant parameters.	2D grayscale image with foreground discernible from background.	
Visualisation	Please make a widget that returns the FFT spectrum of an image as the absolute logarithm of the Fourier transform magnitude. Ensure that the DC component is at the center of the image. Use reflection padding and apodization to reduce artifacts due to discontinuities at the image borders. This widget should work on arbitrary nD images.	Load a 2D or 3D monochrome image.	

Type	Prompts:	Dataset or Image to use:	Notes:
	Please make a widget to rotate the hue of an RGB image and adjust the gamma for the luminance and the gamma for saturation. The default for the angle and gamma values is 1.0. The range of the gamma values is: [0.01, 10].	Load the 'Astronaut' RGB image.	
	Please write a widget that color projects a 3D stack along the Z axis. The hue of the projected pixel is proportional to the depth of the voxel of max intensity, the luminance is proportional to that max intensity, and the saturation is proportional to the contrast between the max intensity and the average intensity.	Load the 'Cells (3D + 2C)' built-in napari sample dataset. Keep the nuclei channel	
	Please create a widget that takes a 2D image as input and splits it into k*k tiles, where k is a parameter in the range [4, 256], snap this parameter to the closest common divisor of both x and y dimensions. For each tile, compute the entropy. The output should be an image of the same dimensions as the input, which is obtained by upscaling the image of entropies (one pixel per tile) using a parameterizable interpolation method. The tile size should be a parameter of the widget.	Load a 2D monochrome image, for example 'camera'.	It is important that the tile size k divide the image dimensions!
	Please make a widget that takes a 2D single-channel image and returns an RGB image in which the hue is proportional to the orientation of the local gradient, the saturation is inversely proportional to the Hessian determinant, and the luminance is proportional to the magnitude of the gradient.	2D RGB image.	
Generation	Please create a widget that generates a random 3D image of given dimensions using Gaussian noise of provided variance and mean value.		
	Create a 2D+t grayscale movie with n Gaussian particles. Each particle performs a random walk with a fixed step size but a random orientation. Parameters include the image dimensions, number of frames, n (number of particles), sigma (for the Gaussian particles), and step size. Don't limit the number of frames, blobs, or image dimensions in the UI. Make sure to write optimised code.		
	Please create a napari widget that takes two image layers and returns a 3D image stack of n images where each 2D image corresponds to a linear blending of the two images.	Two 2D grayscale images.	