lines = MapIndexed[ {GrayLevel@RandomReal[], Line[{#2 - 2 {Cos[#1], Sin[#1]}, #2 + 2 {Cos[#1], Sin[#1]}}]} &, Reverse /@ Transpose@ImageData@gof, {2}];

Transpose a 2×3 matrix:

|  |  |
| --- | --- |
| [In[1]:=](javascript:input('i_1')) | [lick for copyable input](javascript:input('i_1')) |

|  |  |  |
| --- | --- | --- |
| Out[1]= http://reference.wolfram.com/language/ref/Files/Transpose.en/O_1.gif | |  |
|  | |  |
| |  |  | | --- | --- | | [In[1]:=](javascript:input('i_1')) | [lick for copyable input](javascript:input('i_1')) |  |  |  | | --- | --- | | Out[1]= | http://reference.wolfram.com/language/ref/Files/ImageData.en/O_1.gif | | |  |
| [In[1]:=](javascript:input('i_1')) | [lick for copyable input](javascript:input('i_1')) |

|  |  |
| --- | --- |
| Out[1]= | http://reference.wolfram.com/language/ref/Files/Reverse.en/O_1.gif |
|  |  |

|  |  |
| --- | --- |
| [In[1]:=](javascript:input('i_1')) | [lick for copyable input](javascript:input('i_1')) |

|  |  |
| --- | --- |
| Out[1]= | http://reference.wolfram.com/language/ref/Files/MapIndexed.en/O_1.gif |

### [Basic Examples  (2)](http://reference.wolfram.com/language/ref/Graphics.html)

Use lines, polygons, circles, etc. to build up a graphics image:

|  |  |
| --- | --- |
| [In[1]:=](javascript:input('i_1')) | [lick for copyable input](javascript:input('i_1')) |

|  |  |
| --- | --- |
| Out[1]= | http://reference.wolfram.com/language/ref/Files/Graphics.en/O_1.gif |
|  |  |

|  |  |
| --- | --- |
| [In[1]:=](javascript:input('i_1')) | [lick for copyable input](javascript:input('i_1')) |

|  |  |
| --- | --- |
| Out[1]= | http://reference.wolfram.com/language/ref/Files/Line.en/O_1.gif |

|  |  |
| --- | --- |
| [In[1]:=](javascript:input('i_7')) | [lick for copyable input](javascript:input('i_7')) |

|  |  |
| --- | --- |
| Out[1]= | http://reference.wolfram.com/language/ref/Files/ImageEffect.en/O_4.gif |

|  |  |
| --- | --- |
| [In[1]:=](javascript:input('i_1')) | [lick for copyable input](javascript:input('i_1')) |

|  |  |
| --- | --- |
| Out[1]= | http://reference.wolfram.com/language/example/Files/CreateArtisticandPhotographicEffects.en/O_1.gif |

|  |  |
| --- | --- |
| [In[1]:=](javascript:input('i_7')) | [lick for copyable input](javascript:input('i_7')) |

|  |  |
| --- | --- |
| Out[1]= | http://reference.wolfram.com/language/ref/Files/ImageCompose.en/O_4.gif |

Diffusing an angiography image:

|  |  |
| --- | --- |
| [In[1]:=](javascript:input('i_1')) | [lick for copyable input](javascript:input('i_1')) |

|  |  |
| --- | --- |
| Out[1]= | http://reference.wolfram.com/language/ref/Files/CurvatureFlowFilter.en/O_1.gif |

Conclusion Interestingly, Vincent van Gogh's color selection is similar to the median filter result, I was able to convert color of nature to interesting scheme with filtering function, but he has the ability to do that. My filtering function is a result of many Python programmers' hard work. I borrowed a lot of whole or part of that, some times, I tried them all(it didn't work well) The part I directly used from the web, I wrote website address at the first line. The place I learn from filtering function, I mentioned at the first part.