In [1]:
```
#imports
from multiResolution_textureSynthesis import *
from makeGif import *
```

This part requires your input! :)

So, what are all those parameters anyway? Glad you asked!

- **exampleMapPath** - a string with a path to the example image that you want to generate more of!
- **child and parent kernel sizes** - how big is a 'search' window when looking for the best match in the current level (child) and previous level (parent) of the image pyramid. If not sure, keep as default 5 for child and 3 for parent :)
- **saveImgsPath** - a path where you want your output image(s) to be saved to! (the algorithm will also create a txt file with your parameters, so you don't forget the setting you used for different images ^^)
- **saveGifPath** - a path where you want your gif to be saved to
- **pyramidLevels** - how many resolution levels the generation will go through (put 'None' if you want to have the full image pyramid going all the way from single pixel to full image)
- **pyramidType** - gaussian or laplacian

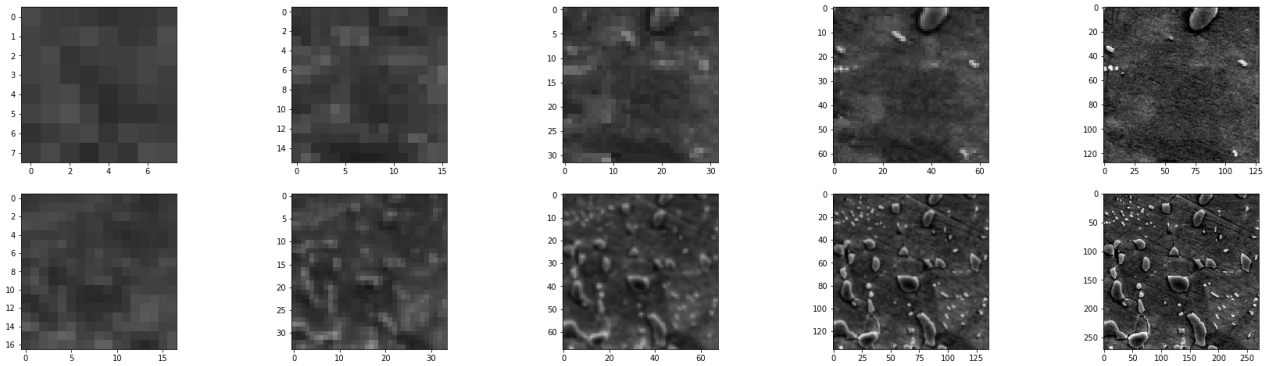
And...that's all! Have fun :)

In [2]:
```
parms = {
  "exampleMapPath": "imgs/8.jpg",
  "outputSize": [128, 128],
  "child_kernel_size": 5,
  "parent_kernel_size": 3,
  "saveImgsPath": "out/33/",
  "saveGifPath": "out/out33.gif",
  "pyramidLevels": 4,
  "pyramidType": "gaussian"
}

#user example will always be treated as level 0 (pass it to 'userExample' parameter)
userExample = {
    "userExamplePath": "userExample_4.jpg"
}
```

In [3]:
```
#run the texture synthesis
multiResolution_textureSynthesis(parms, userExample = None)

#Top row all generated levels
#Bottom row all example levels
```



None

Make a GIF!

In [4]:
```
#make a gif
makeGif(parms["saveImgsPath"], parms["saveGifPath"], frame_every_X_steps = 2, repeat_en
```

Code by Anastasia Opara (www.anastasiaopara.com)

Based on “Fast Texture Synthesis using Tree-structured Vector Quantization” and “Multiresolution Sampling Procedure for Analysis and Synthesis of Texture Images” papers