BINARY GENETIC PROGRAMMING BASED TEXTURE SEGMENTATION
WHAT IS TEXTURE?

• There is no definite answer, but an possible definition could be that a texture is a kind of visual information that comes in different shapes and sizes yet we should be able to differentiate between different textures.

• Texture is two dimensional image information formed by prominent variation in gray levels between neighboring pixels. The variation possibly is periodic. This variation can be considered as texture regularity or texture homogeneity.
Texture is essential to any vision system. Animals for example could detect the presence of a lion using its unique textures. Also Mars probe Pathfinder uses texture to detect sky, dust and other materials in the space.

Textures are homogenous visual patterns such as grass, wood, and sand. The uniformity in patterns is essential for scene understanding.

This study aims at using multi classifier genetic programming algorithm for texture classification in binary form.

Classification problems require searching through many partial candidate solutions to find a good solution. Genetic Programming allows finding a solution in a search space quickly and automatically.

The choice of genetic programming was due to its ability to solve diverse range of complex problems. It works by developing a population of "computer programs" are evolved over a number of generations. Genetic Programming was very successfully in tasks like classification problems, and image analysis.

The process contains two steps, first texture feature extractions, and then using classification algorithm to classify textures.
Texture analysis provides an intuitive presentation of textures, and produce texture features that are description of texture.

There are two approaches for texture analysis:
- **Macro**: views textures as composition of textural primitives, but to do that it needs to identify textural primitives and relations among these primitives.
- **Micro**: Measures textures without identifying textural primitives.
Texture segmentation is a complex texture analysis task that partitions an image into regions based on differences in texture appearance. It does not only identify the boundaries of each region, but also the textures inside each region.

The main applications of texture segmentation is separating objects from background. It's also worth noting that image segmentation methods such as edge detection, thresholding, and region growing are not applicable to texture segmentation.

Texture segmentation could be supervised and unsupervised, with unsupervised approach there is no need to know numbers of textures on the scene or what do they look like, but the problem with unsupervised approach is adding extra dimensionality of complexity since the number of textures is unknown.
SEGMENTATION METHODS

- Boundary Image Segmentation: Identifies the usual noncontinuous and unclosed boundaries between two textures.

- Region Based Segmentation: Concerned with identifying texture regions based on adjacent parts of regions that have different textures.

- Split-and-merge:
  - It works by splitting image into quarters, and then analyze every two quarters, if two quarters are different, then it's going to split again recursively otherwise the two quarters will be merged again.
  - Split-and-merge with sliding window: Slides a window over the whole image to sample small regions that might overlap. Segmentation is then done by classification voting strategy, in which regions are linked to specific classes. Due to overlapping between regions, a pixel might be classified into different classes, for each of those a voting scheme is used to choose a single class for each pixel.
  - Split and merge is known for its low computational overhead, but it produces blocky boundaries that requires further smoothing.
The key difference between evolutionary techniques is the representation used to model potential solutions.

Individuals in the GP population are software written for example in languages like C++ or Java that can perform independently.

Operators such as crossover and mutation are used to generate new programs. In crossover two programs exchange part of their program code to make two new programs. While in mutation part of one program’s code is replaced by newly randomly created to create a new one.

Mutations with genetic programming are means of searching the entire search space, so it's common to generate low quality solutions due to random nature of mutations.
GENETIC PROGRAMMING: TREE GP

- Operators are drawn from a pool defined by the programmer.
- Arguments are also defined, and they include image features, constant real numbers, etc..
**DATA SETS**

- **Brodatz album** is the most commonly used database for texture segmentation. It is the de-facto standard database for texture-related research. The Brodatz album has very diverse collection of beach sand, grass, cloud and many others.

- **Vistex** was published by Massachusetts Institute of Technology (MIT) for research on image processing, and computer vision. It represents real world conditions such as buildings, and paintings. In comparing Brodatz to Vistex, it was shown that Brodatz has higher classification accuracy than Vistex which in turn might mean that Vistex would complicate the process of finding an accurate classifier.

- **Meastex** is a collection that contains some Brodatz, and Vistex images in addition to some natural and artificial textures images.
Thank You