AImageOperation (abstract class)
- preProcessing() : void
- executeOperation() : void
- postProcessing() : void
- performClamping(Image) : void
- getColumns(double[][]) : int
- getRows(double[][]) : int
- getColumns(int[][]) : int
- getRows(int[][]) : int
- operationType : EOperationType

ImageFilter
- inputImage : Image
- mapFilterTypeKernel : Map<EFilterType, double[][]>
- ImageFilter(Image, EFilterType)
- initMapFilterTypeKernel() : void
- preProcessing() : void
- getPixelValue(int, int, int[][]) : int
- executeOperation() : void
- dotProduct(int[][]) : int
- postProcessing() : void
- validateKernelMatrix(double[][]) : void
- filterType : EFilterType
- operationType : EOperationType

ImageColorTransformation
- inputImage : Image
- eColorTransformation : EColorTransformation
- mapColorTransformationMatrix : Map<EColorTransformation, float[][]>
- ImageColorTransformation(Image, EColorTransformation)
- initMapColorTransformationMatrix() : void
- preProcessing() : void
- executeOperation() : void
- mulTransformationMatrixWithRGB(int, int) : void
- postProcessing() : void
- validateTransformationMatrix(float[][]) : void
- operationType : EOperationType

ImagePatternGeneration
- preProcessing() : void
- executeOperation() : void
- postProcessing() : void
- operationType : EOperationType

ImageColorDensity
- inputImage : Image
- ditherAlgorithm : IDither
- ImageColorDensity(Image, EColorDensity, IDither)
- preProcessing() : void
- executeOperation() : void
- postProcessing() : void
- operationType : EOperationType
- colorDensity : EColorDensity

ImageChunking
- inputImage : Image
- chunkingStrategy : IChunkingStrategy
- ImageChunking(Image, IChunkingStrategy)
- preProcessing() : void
- executeOperation() : void
- postProcessing() : void
- operationType : EOperationType

IAlgorithm (interface)
- execute(Image) : void

IPatternGenStrategy
- patternGenStrategy : EPatternGenStrategy

IChunkingStrategy
- chunkingStrategy : EChunkingStrategy

IDither

TextPattern
- execute(Image) : void
- patternGenStrategy : EPatternGenStrategy

ClosestColor
- execute(Image) : void
- patternGenStrategy : EPatternGenStrategy

Pixelation
- square : int
- Pixelation(int)
- execute(Image) : void
- chunkingStrategy : EChunkingStrategy

Mosaic
- seeds : int
- Mosaic(int)
- execute(Image) : void
- chunkingStrategy : EChunkingStrategy

FloydSteinbergDither
- isEssence : boolean
- totalColors : int
- FloydSteinbergDither(boolean, int)
- increasePixelValueBy(int, int, int[][], float) : void
- findClosesPaletteColor(int, int) : float
- execute(Image) : void

EnhancedImageBuilder
- operations : List<AImageOperation>
- imgFilePath : String
- image : Image
- EnhancedImageBuilder(String)
- readImage(String) : Image
- addOperation(AImageOperation) : EnhancedImageBuilder
- addFilter(EFilterType) : EnhancedImageBuilder
- addColorTransformation(EColorTransformation) : EnhancedImageBuilder
- addColorDensity(EColorDensity, boolean, int) : EnhancedImageBuilder
- addChunkingStrategyMosaic(int) : EnhancedImageBuilder
- addChunkingStrategyPixelation(int) : EnhancedImageBuilder
- build() : EnhancedImageBuilder
- saveImage(String) : EnhancedImageBuilder

EOperationType
- FILTER
- COLOR_TRANSFORMATION
- COLOR_DENSITY
- CHUNKING
- PATTERN_GEN
- EOperationType(String)
- values() : EOperationType[]
- valueOf(String) : EOperationType
- value : String

Image
- Image(int[][][])
- Image(String)
- getChannelMatrix(Channel) : int[][]
- convertImageInChannelMap() : Map<Channel, int[][]>
- convertChannelMap2Image(Map<Channel, int[][]>, Image) : void
- rgbMatrix : int[][][]
- imageWidth : int
- imageHeight : int

EColorTransformation
- GREY_SCALE
- SEPIA_TONE
- EColorTransformation(String)
- values() : EColorTransformation[]
- valueOf(String) : EColorTransformation
- value : String

EPatternGenStrategy
- CLOSEST_COLOR
- TEXT_PATTERN
- EPatternGenStrategy(String)
- values() : EPatternGenStrategy[]
- valueOf(String) : EPatternGenStrategy
- value : String

EChunkingStrategy
- MOSAIC
- PIXELATION
- EChunkingStrategy(String)
- values() : EChunkingStrategy[]
- valueOf(String) : EChunkingStrategy
- value : String

EFilterType
- BLUR_FILTER
- SHARP_FILTER
- EFilterType(String)
- values() : EFilterType[]
- valueOf(String) : EFilterType
- value : String

EColorDensity
- REDUCE
- EColorDensity(String)
- values() : EColorDensity[]
- valueOf(String) : EColorDensity
- value : String

ImageUtilities
- readImage(String) : int[][][]
- getWidth(String) : int
- getHeight(String) : int
- writeImage(int[][][], int, int, String) : void

FileUtils
- isFileExist(String) : boolean
- getFileNameFromPath(String) : String
- getDirFromPath(String) : String

BatchController
- main(String[]) : void

Powered by yFiles

Design/Model Changes.

Following are the changes that are made:

- Concept of Inheritance is used properly; all the operations (e.g. Filtering, Color Transformation, Color Reduce, Image Chunking and Image Pattern Generation) drive from common abstract AImageOperation.

- Create enums for:

  - Filtering type i.e. EFilterType (Blur Filter, Sharp Filter), in future if any new filter is introduce we can easily update our enum.

  - Color Transformation i.e. EColorTransformation (Grey Scale, Sepia Tone)

  - Color Density i.e. EColorDensity (Reduce), currently it only store one value but in future it can be update.

  - Chunking Image i.e. EChunkingStrategy (Mosaic, Pixelation)

  - Pattern Generation i.e. PatternGenStrategy (Closest Color, Text Pattern)

- Created three new interfaces IPatternGenStrategy, IChunkingStrategy and IDither that are drive from one common interface IAlgorithm.

  - TextPattern and ClosestColor algorithms extend IPatternGenStrategy.

  - Pixelation and Mosaic algorithms extend IChunkingStrategy.

  - FloydSteinberg Dithering algorithm extends IDither.

- Categorize execution flow into three different stages i.e. Pre Processing, execute Operation, Post Processing

- Make the entire field private final.

- Use builder pattern to apply different combination of enhancements on same image.

- Use open close principle to avoid any modification in future and encourage extension.

- Use batch controller in the model that will respond to user commands.

Testing Plan

Mock model drives from common abstract i.e. AImageOperation.

The batch controller i.e. BatchController composed from EnhancedImageBuilder class (Mock model) that performs image operations:

1- Image Filter

2- Color Transformation
3- Color Density
4- Image Mosaic
5- Image Pixelation
6- Pattern Generation

Input	Operations	Output
load goat.jpg	blur filter	save blur_filter_goat.jpg
load goat.jpg	Dither	save 8_values_per_channel_with_essence_goat.jpg
load goat.jpg	pixelate 50 pattern	save goat_pattern.txt
load birds.jpg	sepia tone sharp	save birds_multiple_filter.png
load birds.jpg	mosaic	save birds_mosaic.png
load birds.jpg	Pixelate 30	save birds_pixelation.png