Group - 12

DS
PROJECT

PRIYANSHI VERMA-22B1503
SAUJANYA CHANDRA-22B3977
RISHABH BHARDWAJ-22B3962
OVIYA S-22B3918

PROJECT OVERVIEW

The provided dataset consists of 1183 images of plan view of buildings of an architecture company. The following set of tasks have been assigned in this project :

- Shape-based grouping for design insights and standardization
- Classify layout complexity: Low, Medium, High.
- Speed up design process with parameter-based layout retrieval.
- Explore additional image data possibilities for design enhancement.

DATA PREPARATION

- Firstly we removed the duplicate images from the folder by creating contours using layout.
- Hu moments were used to extract distinct images.
- We verified this using siamese model.
- Finally we had 173 unique images

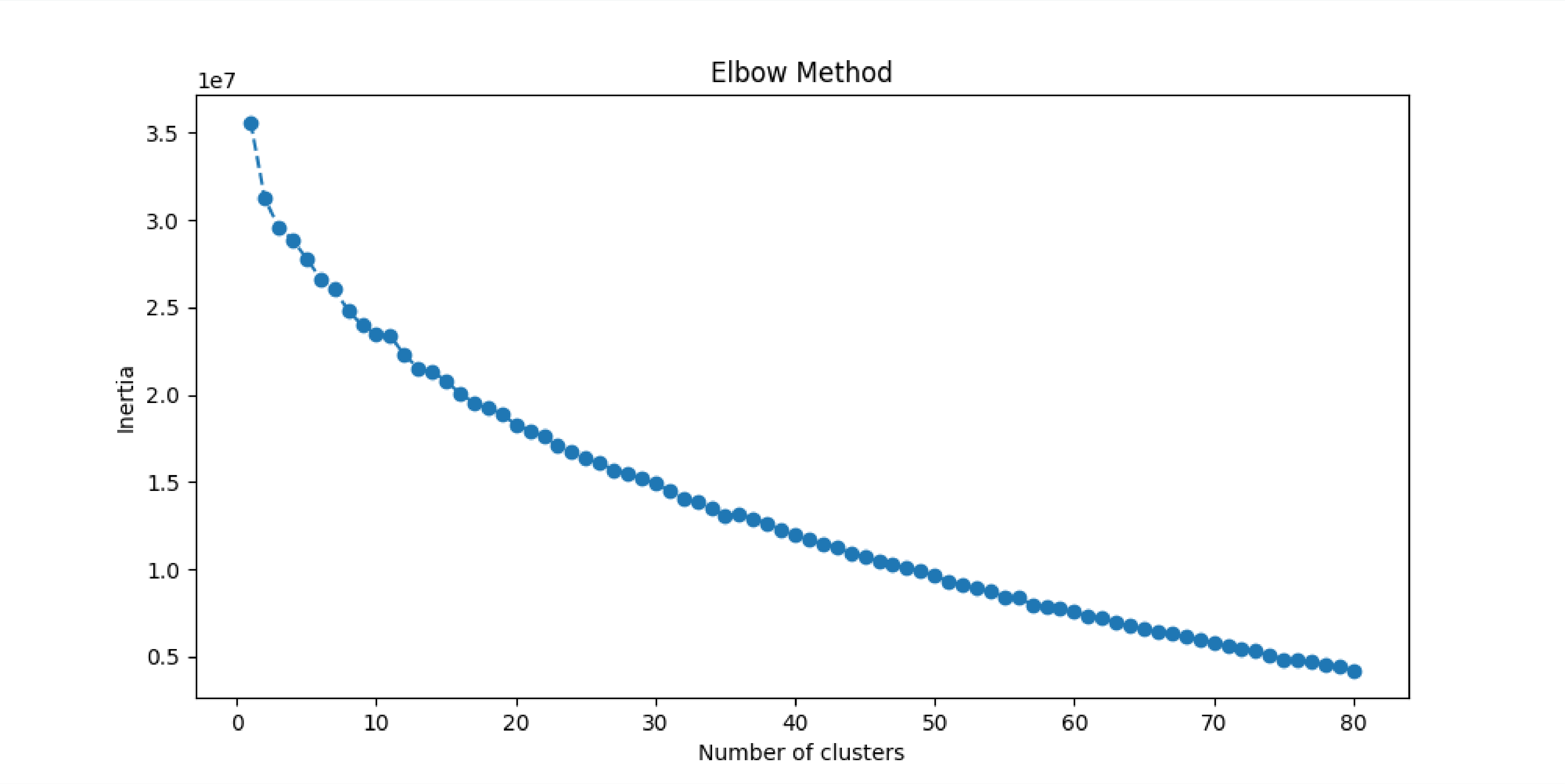
GROUPING DESIGNS INTO FAMILIES

We used 3 methods to divide the given images into groups (Design families):-

1. BASED ON SHAPE SIMILARITY (ELBOW METHOD)
2. BASED ON RATIO OF LAYOUT AREA : TOTAL TIGHT FITTING BOX AREA
3. BASED ON RATIO OF DIMENSION OF TIGHT FITTING BOX

Approach 1: Elbow method

- Plotted inertia against cluster count
- Analyzed the generated graph and chose **35 clusters**.



Depiction of clusters | Cluster 13

1178.jpg

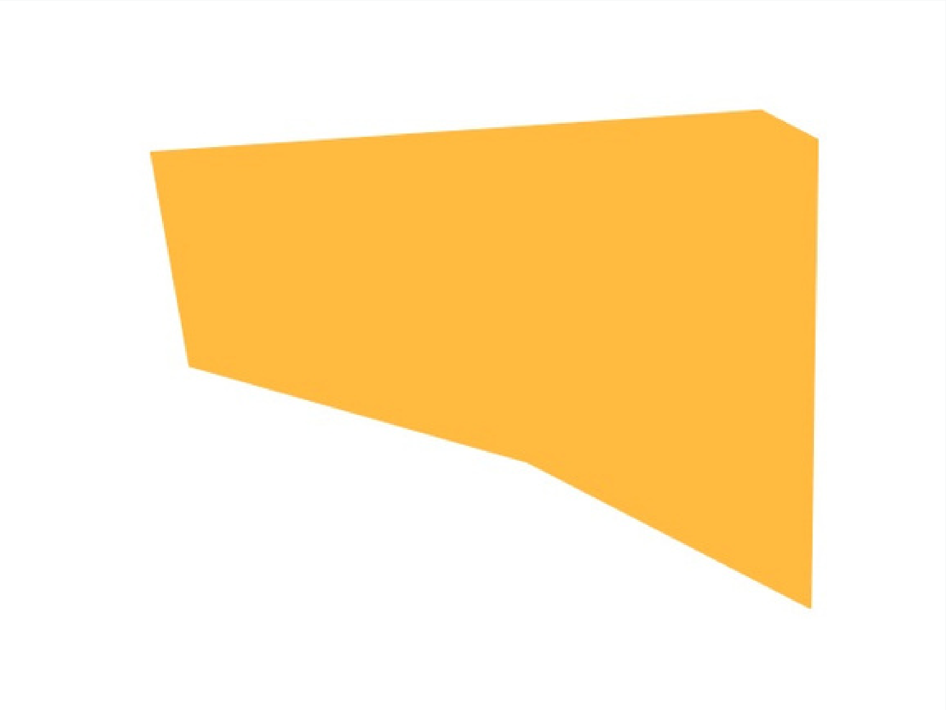
0971.jpg

0896.jpg

Depiction of clusters | Cluster 19

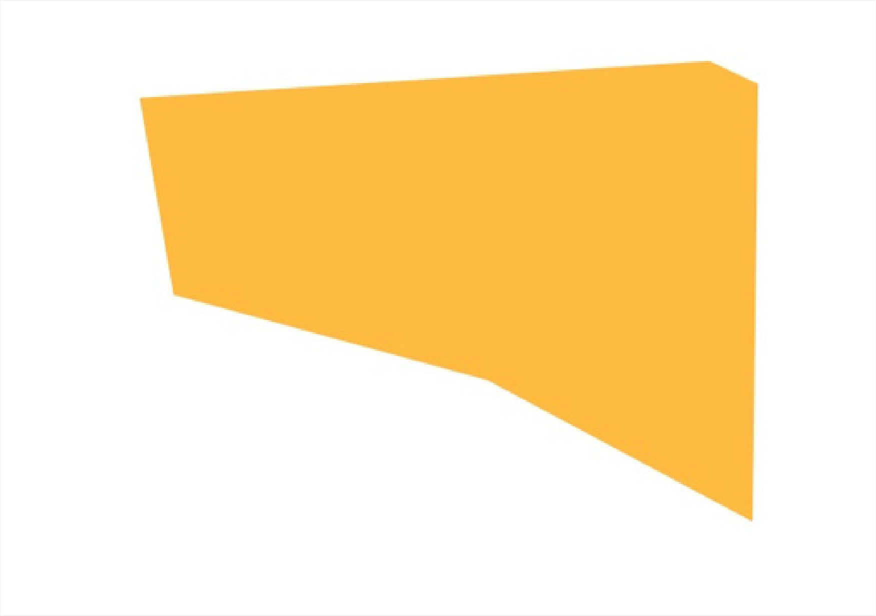
0197.jpg

1019.jpg



0714.jpg

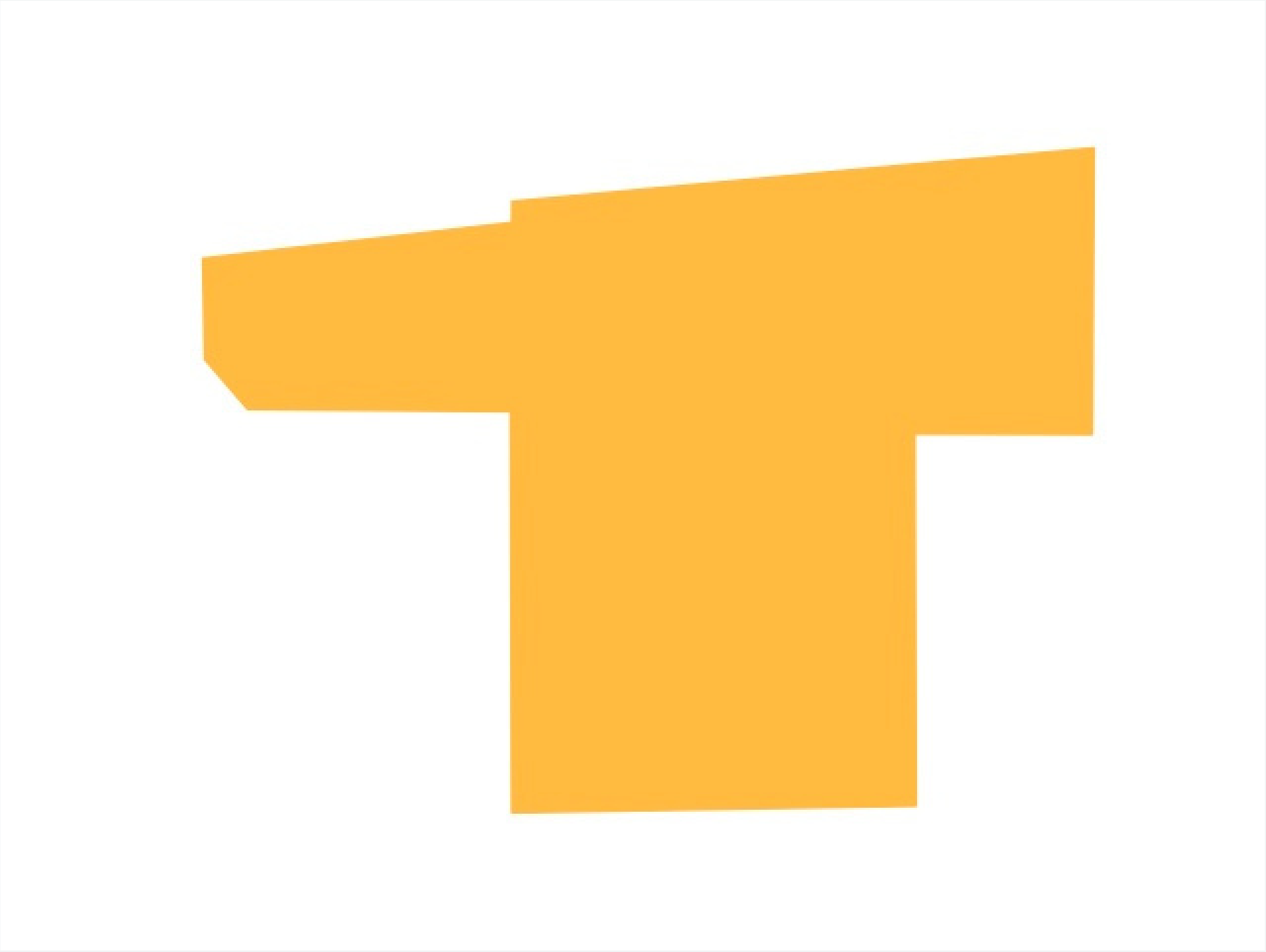
0238.jpg



0288.jpg

1083.jpg

Depiction of clusters | Cluster 20

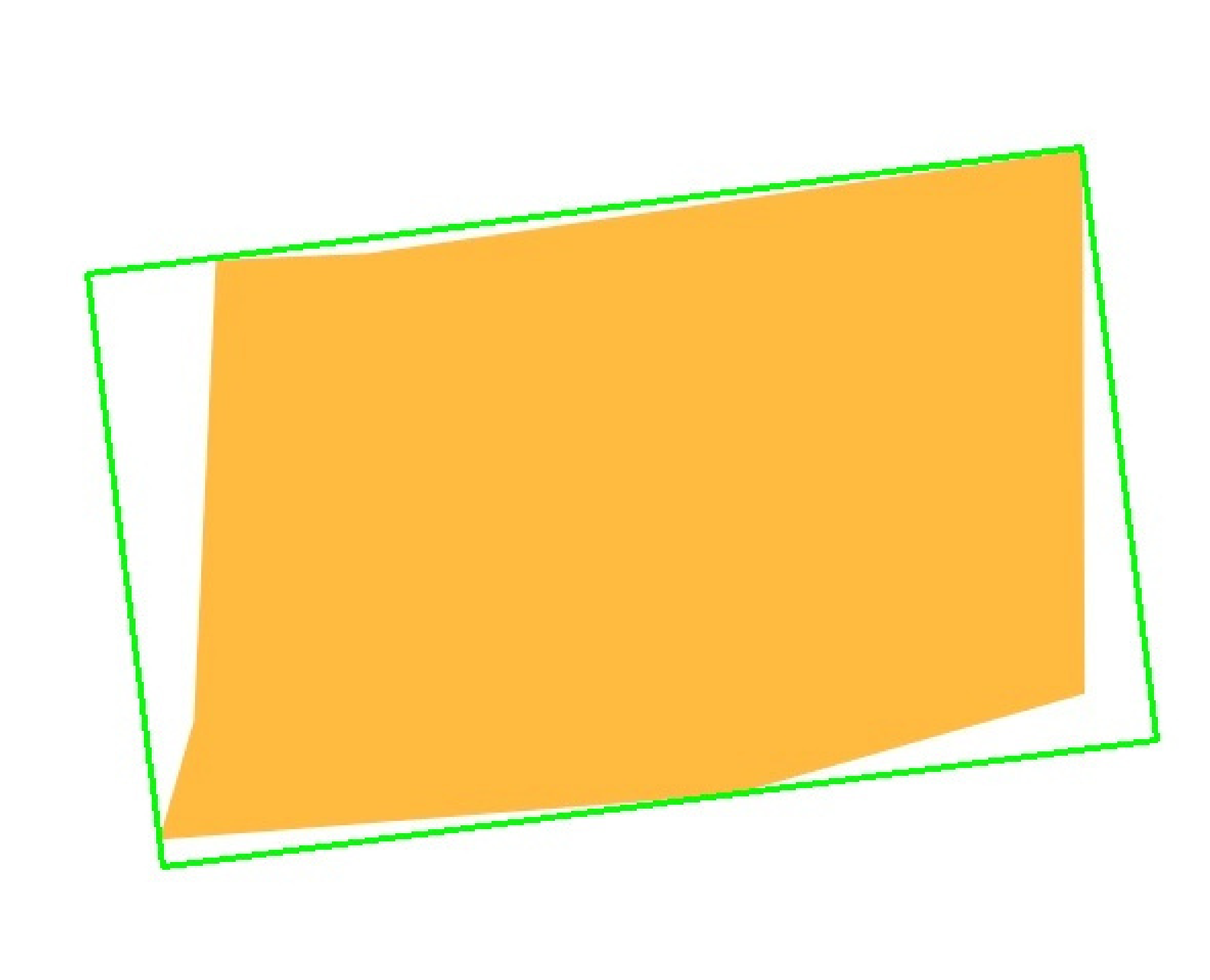


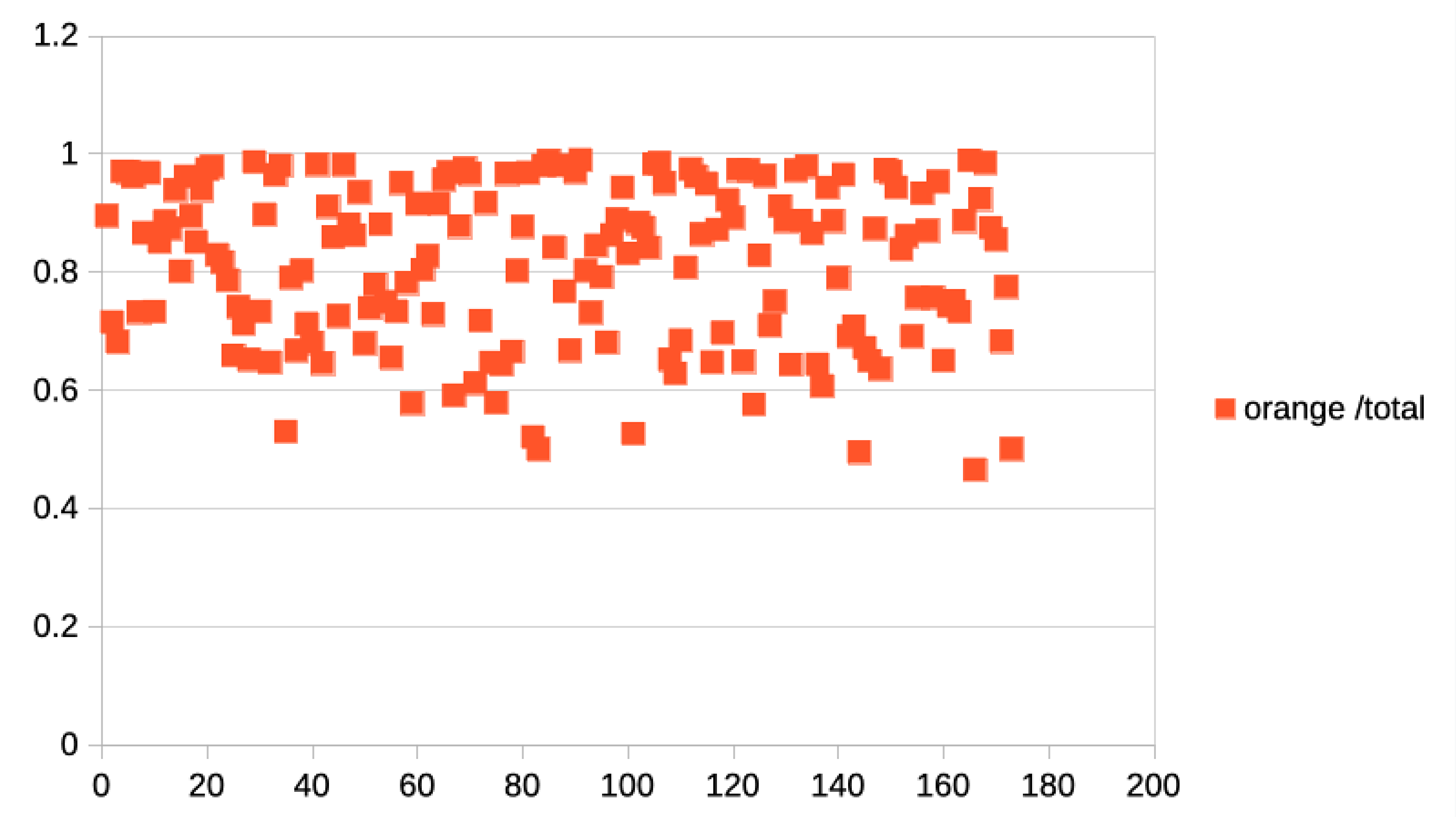
0934.jpg

0466.jpg

Approach 2: Based on ratio of area

- Created tight-fitting bounding box for each layout
- Calculated ratio of layout area : Total box area





Approach 2: Based on ratio of area

- Formed 8 design families according to range of area ratio

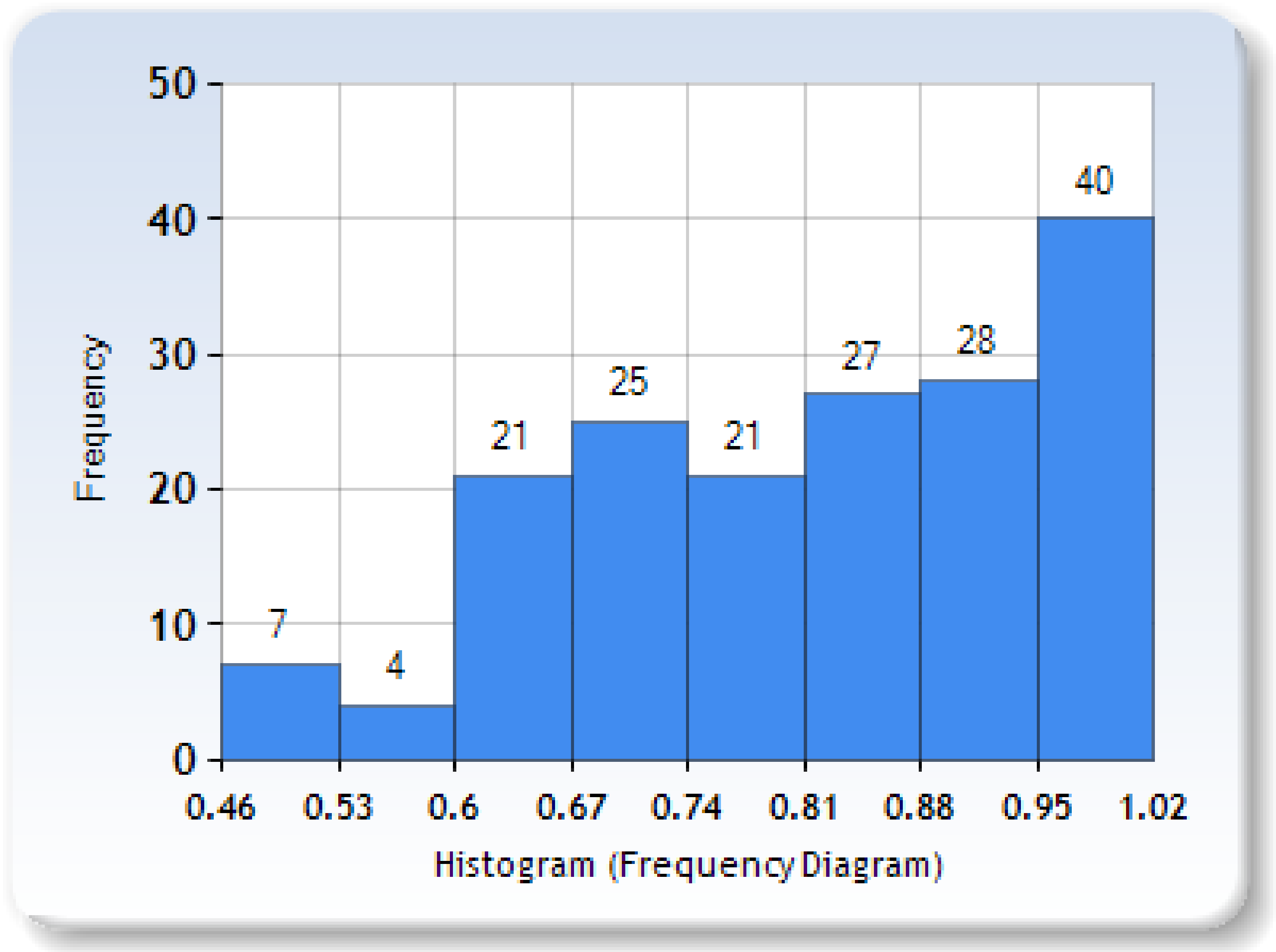
Your Histogram	
Mean	0.81451
Standard Deviation (s)	0.1352
Skewness	-0.50009
Kurtosis	-0.74414
Lowest Score	0.465646
Highest Score	0.989429
Distribution Range	0.52378
Total Number of Scores	173
Number of Distinct Scores	173
Lowest Class Value	0.46
Highest Class Value	1.019999
Number of Classes	8
Class Range	0.07

Frequency Table	
Class	*Count*
0.46-0.529999	7
0.53-0.599999	4
0.6-0.669999	21
0.67-0.739999	25
0.74-0.809999	21
0.81-0.879999	27
0.88-0.949999	28
0.95-1.019999	40

Approach 2: Based on ratio of area

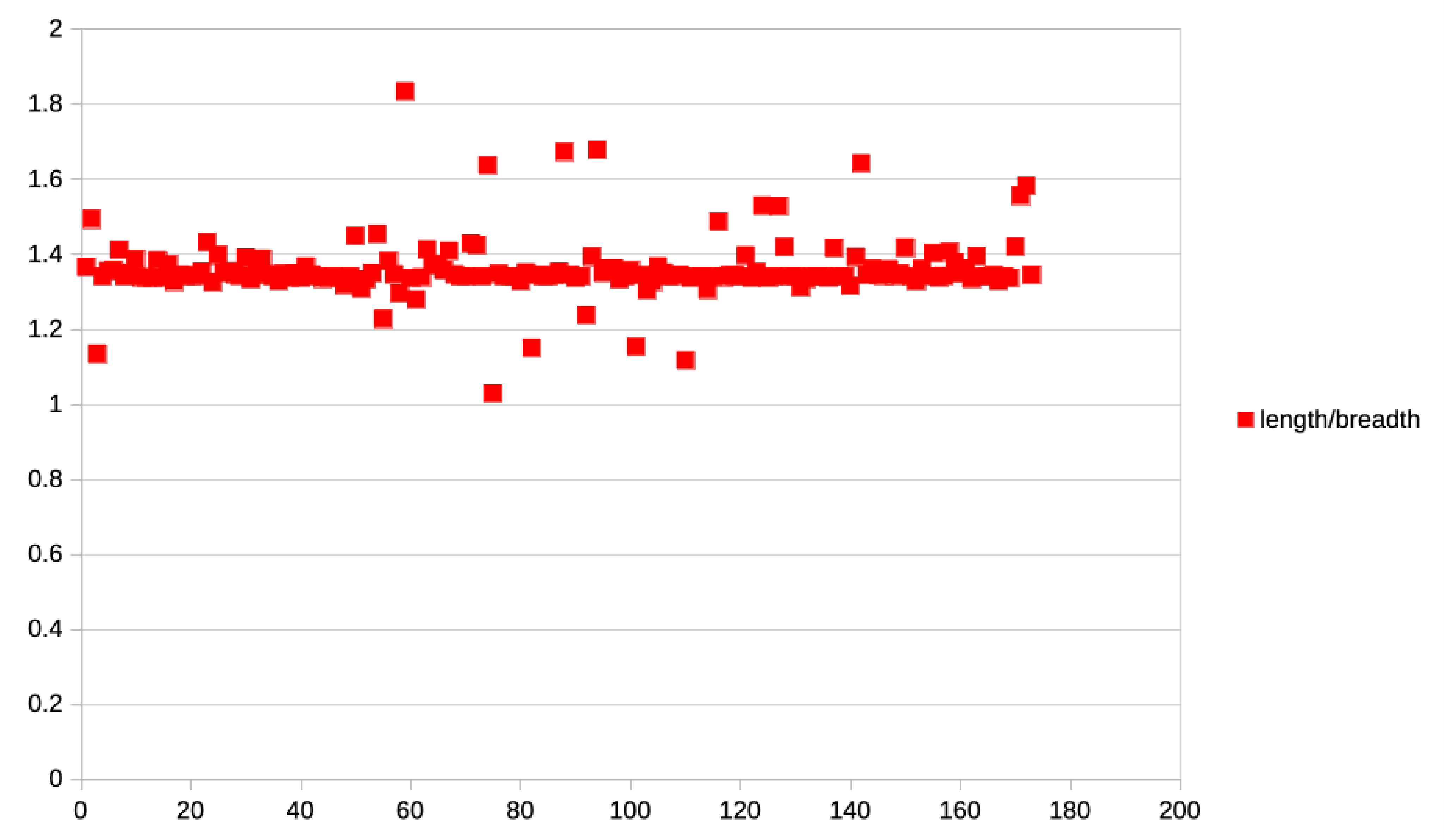
Observations

- Maximum number of layouts found under the range 0.95 - 1.02
- Minimum number of layouts fall under the range 0.53 - 0.6



Approach 3 : Based on ratio of length:breadth of bounding boxes

- Created tight-fitting bounding box for each layout
- Calculated ratio of length:breath



Approach 3 : Based on ratio of length:breadth of bounding boxes

- Formed 6 design families according to range of ratios
- 1-1.21 , 1.22 - 1.32 , 1.33-1.43 , 1.44 - 1.54 , 1.55 - 1.65, 1.66 - 1.87

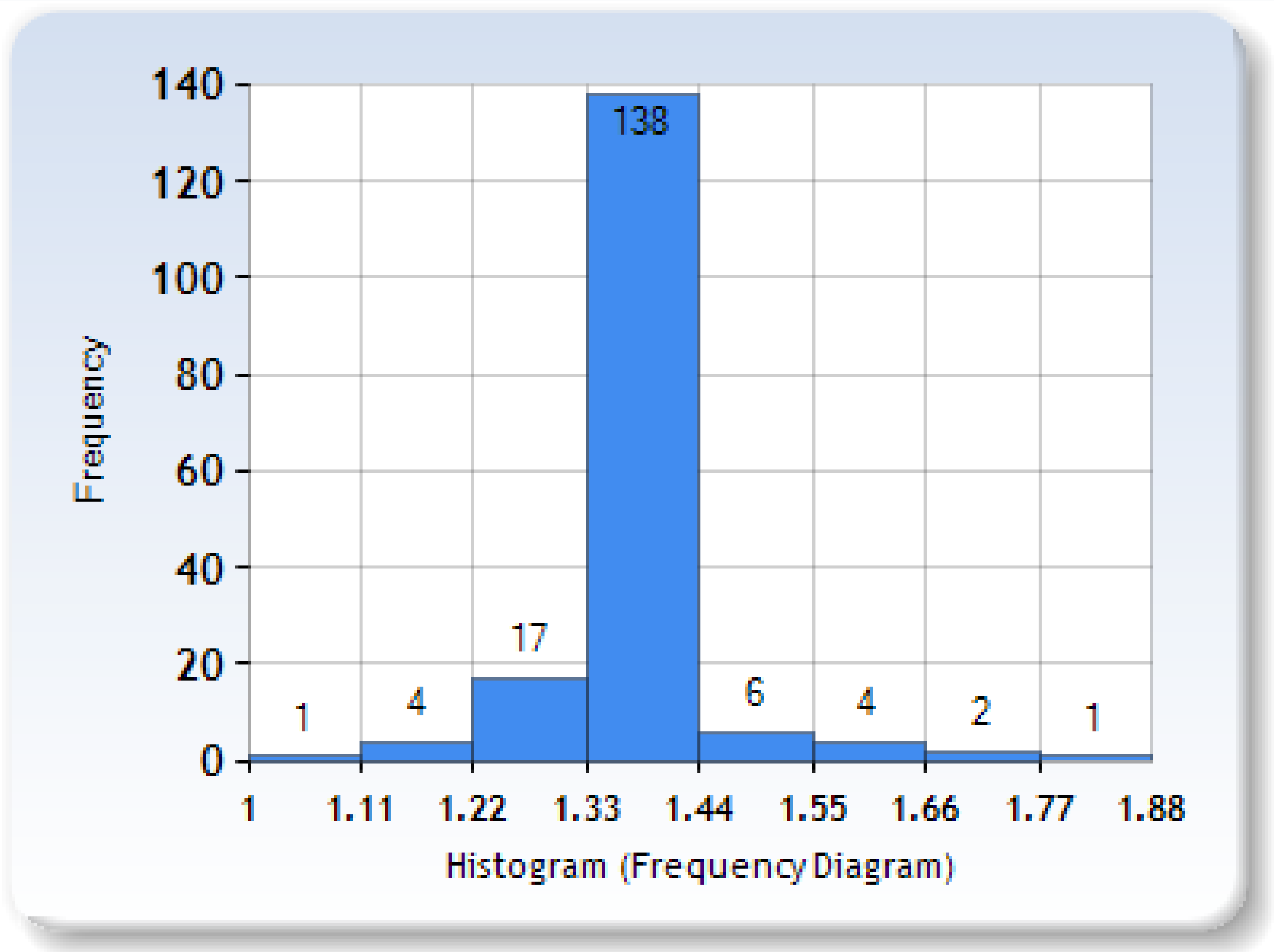
Your Histogram	
Mean	1.36003
Standard Deviation (s)	0.08459
Skewness	1.58161
Kurtosis	10.0529
Lowest Score	1.028815
Highest Score	1.834195
Distribution Range	0.80538
Total Number of Scores	173
Number of Distinct Scores	132
Lowest Class Value	1
Highest Class Value	1.879999
Number of Classes	8
Class Range	0.11

Frequency Table	
Class	*Count*
1-1.109999	1
1.11-1.219999	4
1.22-1.329999	17
1.33-1.439999	138
1.44-1.549999	6
1.55-1.659999	4
1.66-1.769999	2
1.77-1.879999	1

Approach 3 : Based on ratio of length:breadth of bounding boxes

Observations

- Maximum number of layouts found under the range 1.33 - 1.44
- Minimum number of layouts fall under the range 1 .66 - 1.88

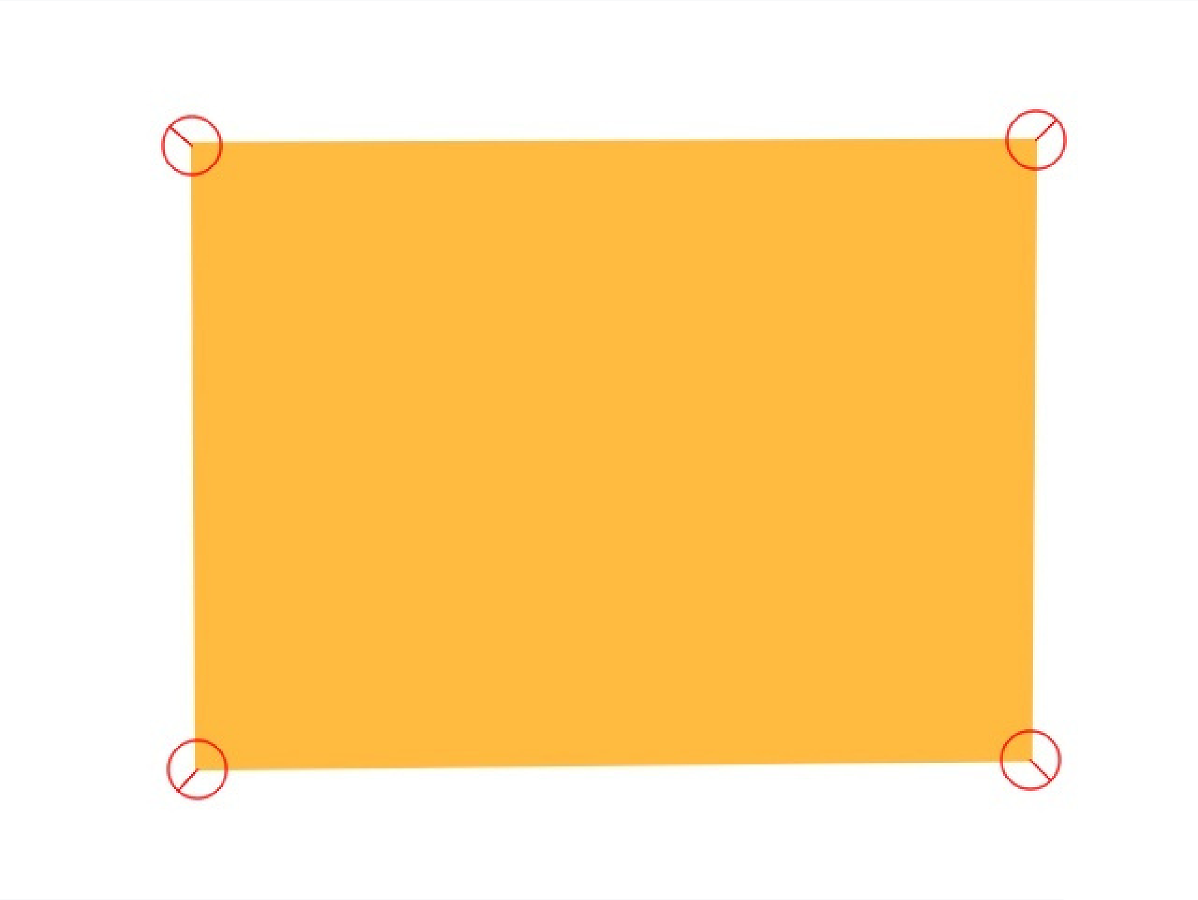


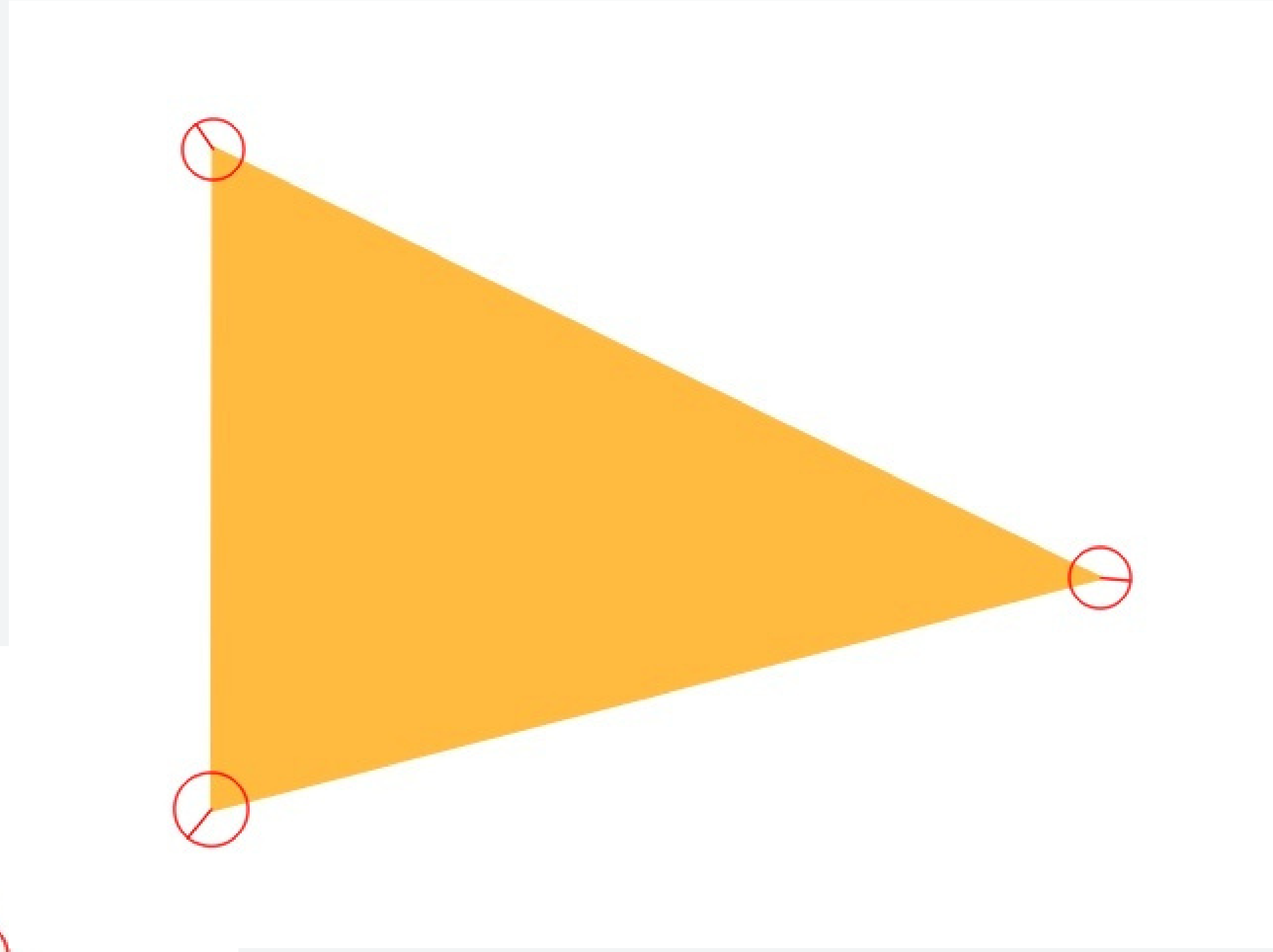
CLASSIFICATION OF COMPLEXITY OF LAYOUTS

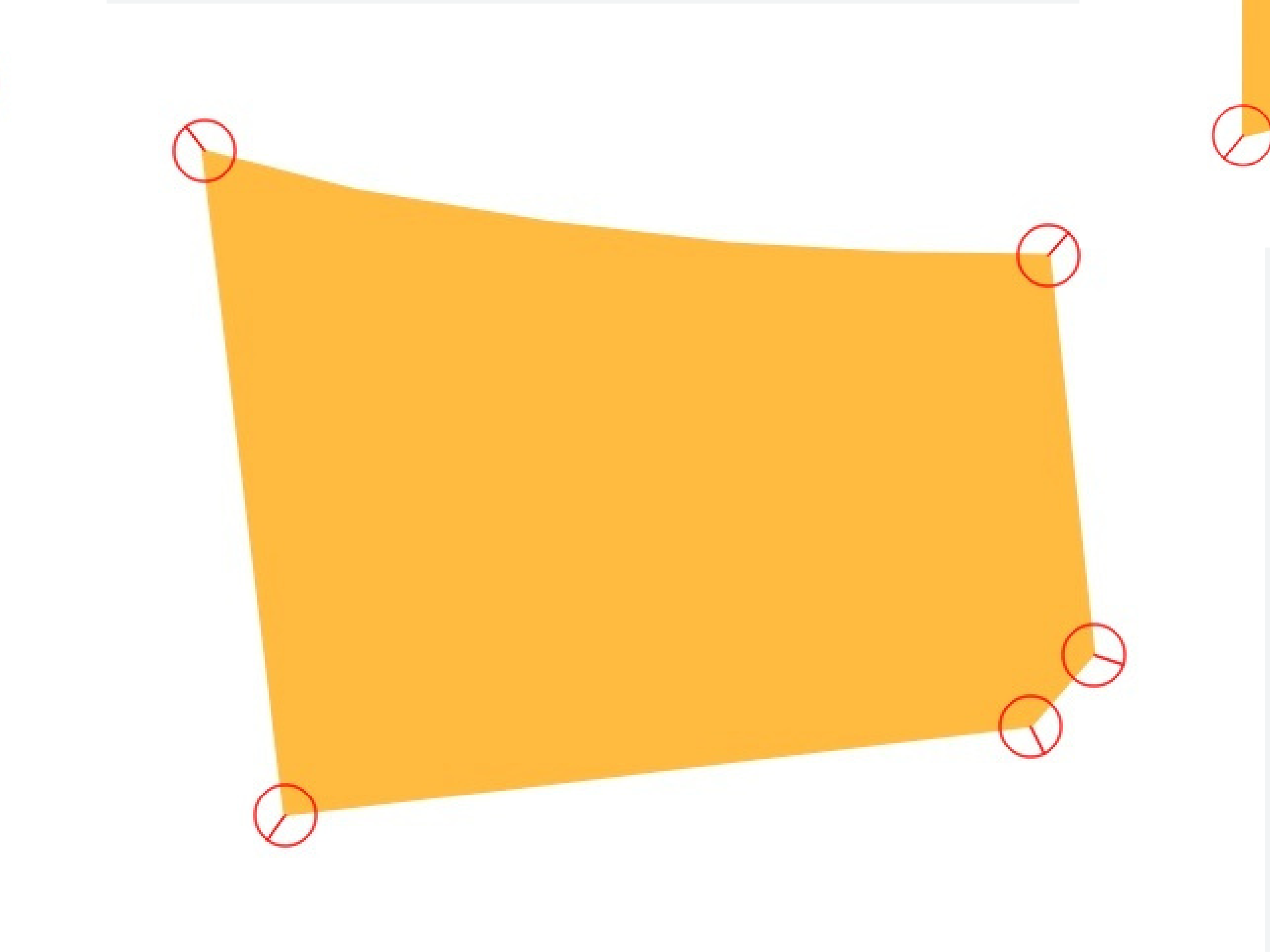
- We have classified the complexity of layouts based on number of corners
- Detected number of corners using MobileNetV2, found range of corners to be 3- 20
- Classification decided:

1. Low complexity - 3 to 5 corners
2. Medium complexity- 6 to 9 corners
3. High complexity - 10 to 20 corners

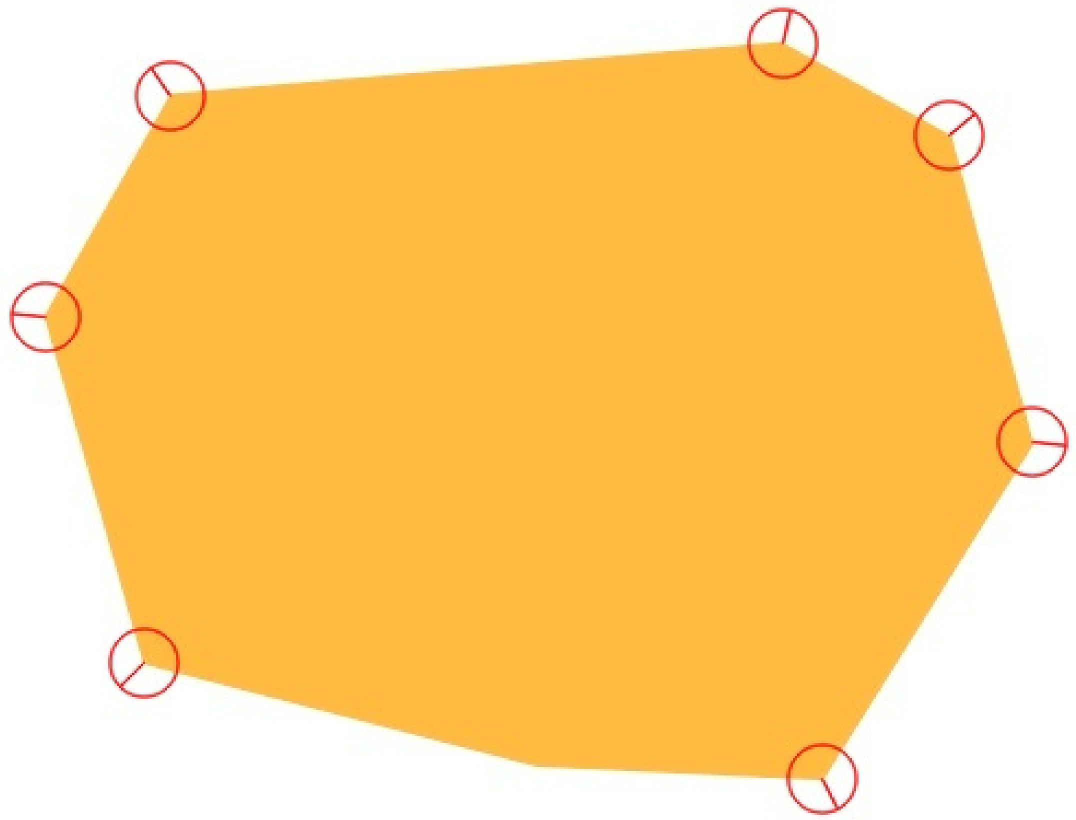
Low complexity
3-5 corners

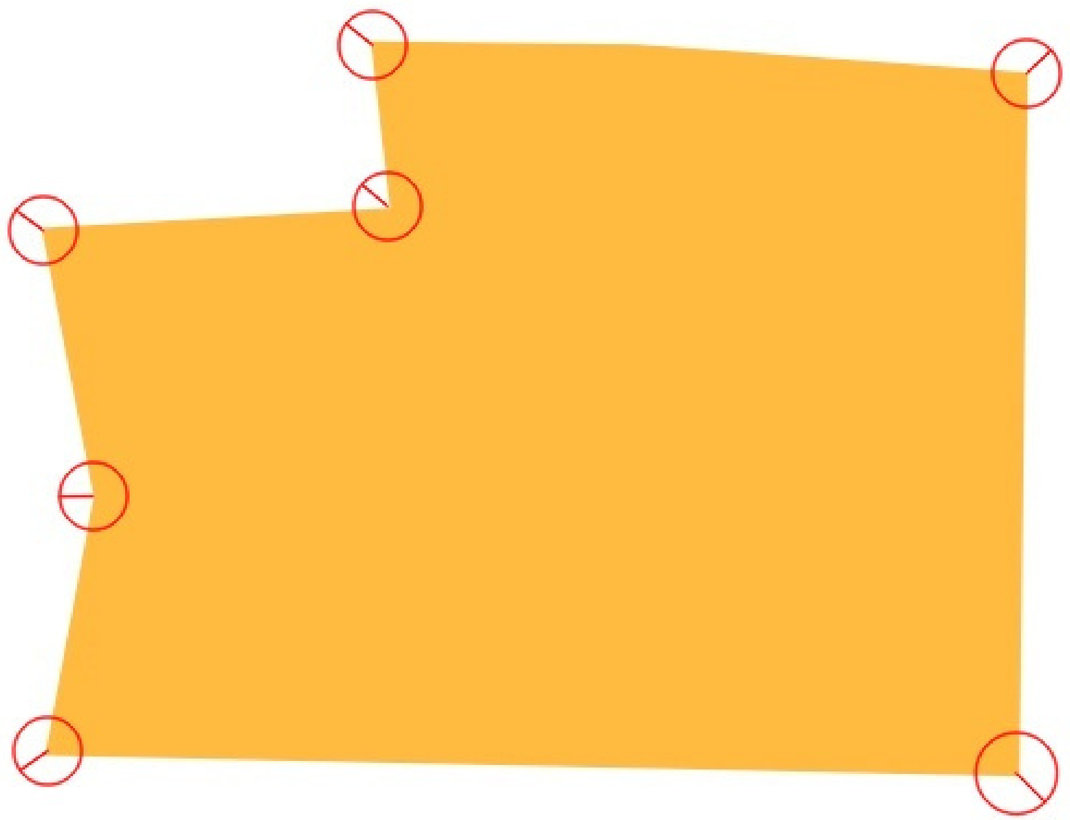


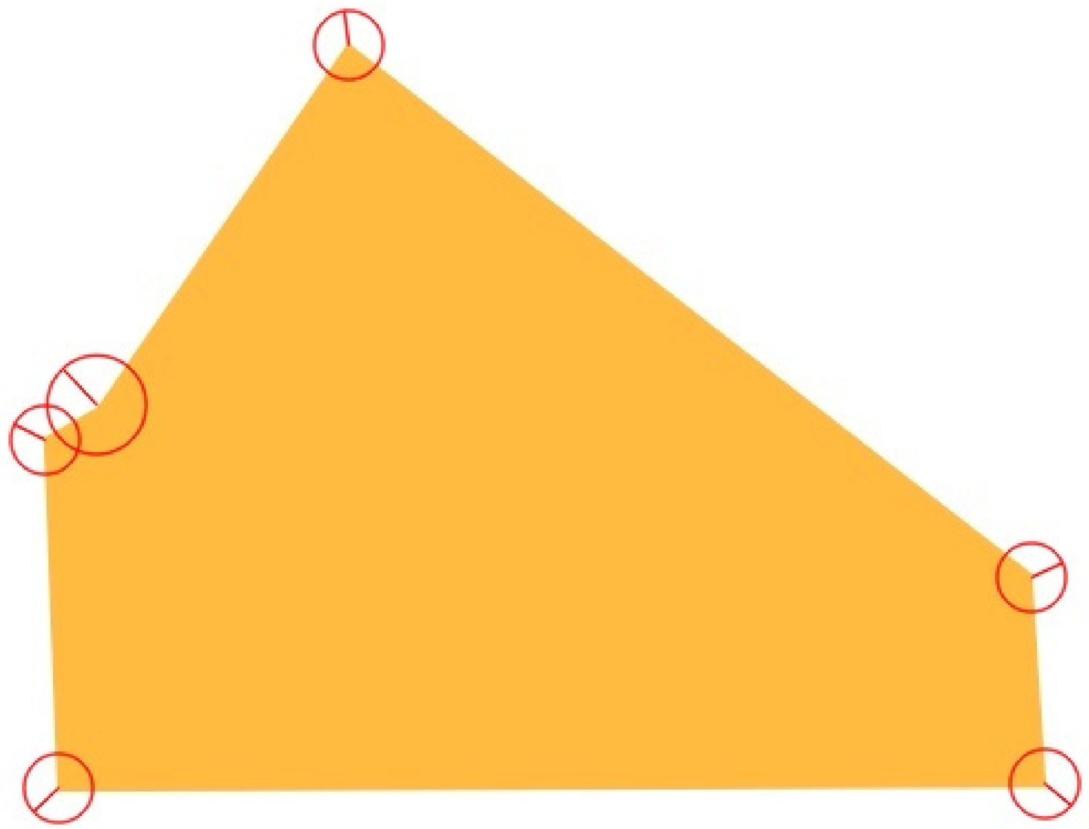




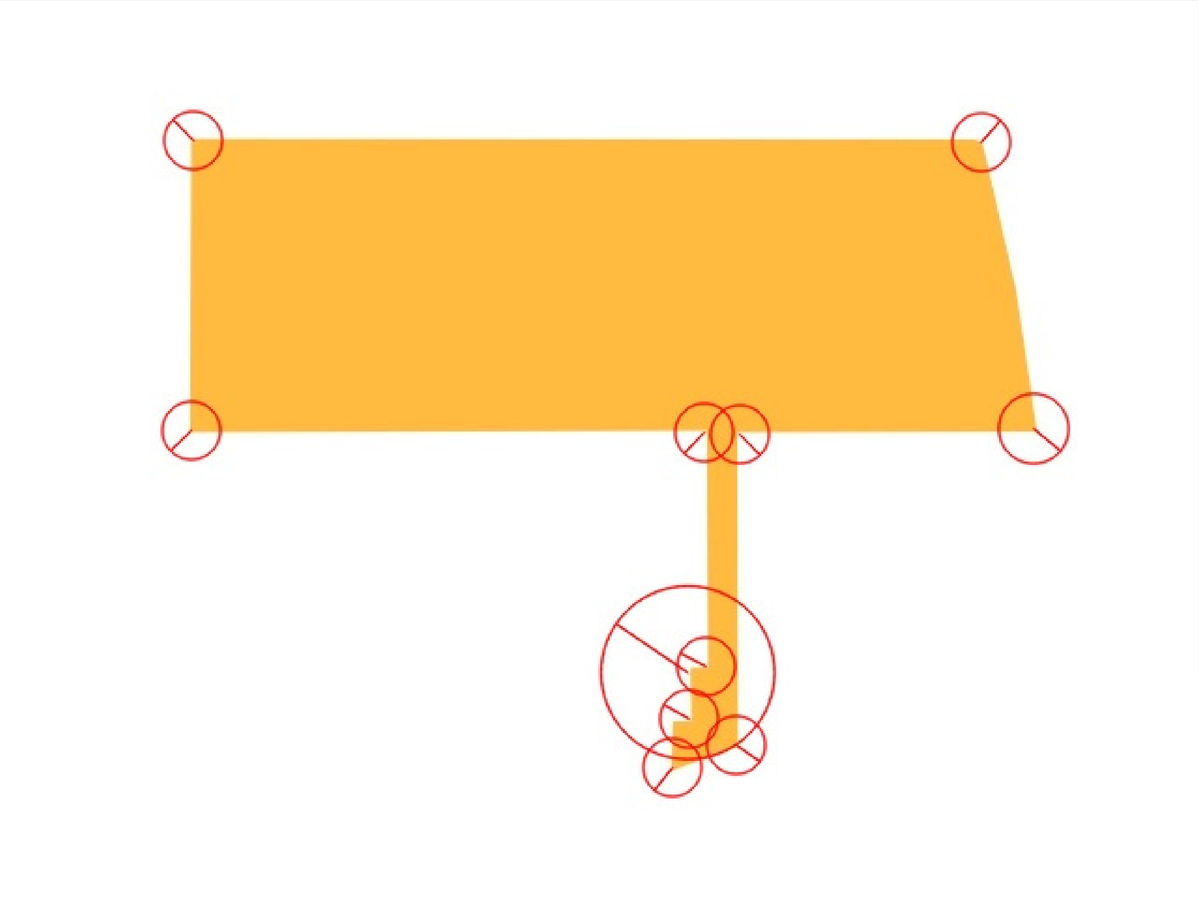
Medium complexity
6 to 9 corners

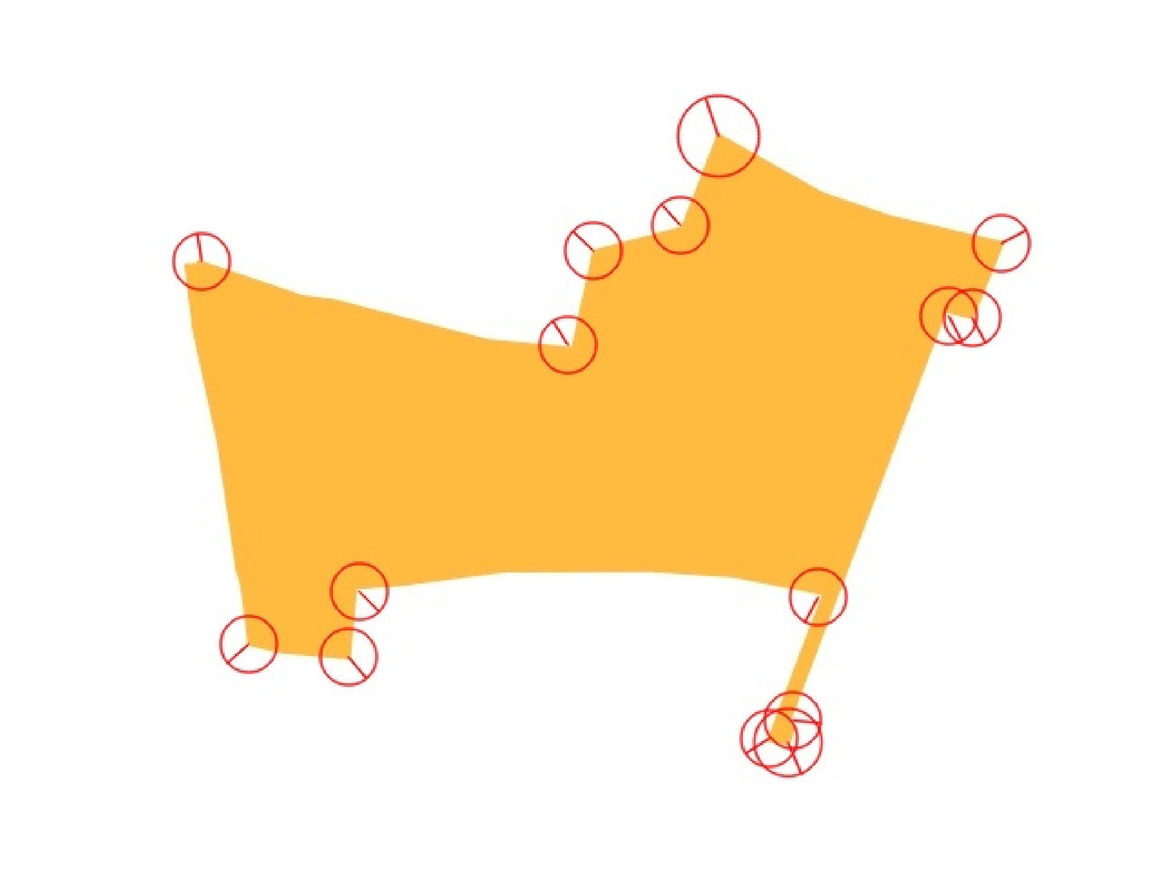


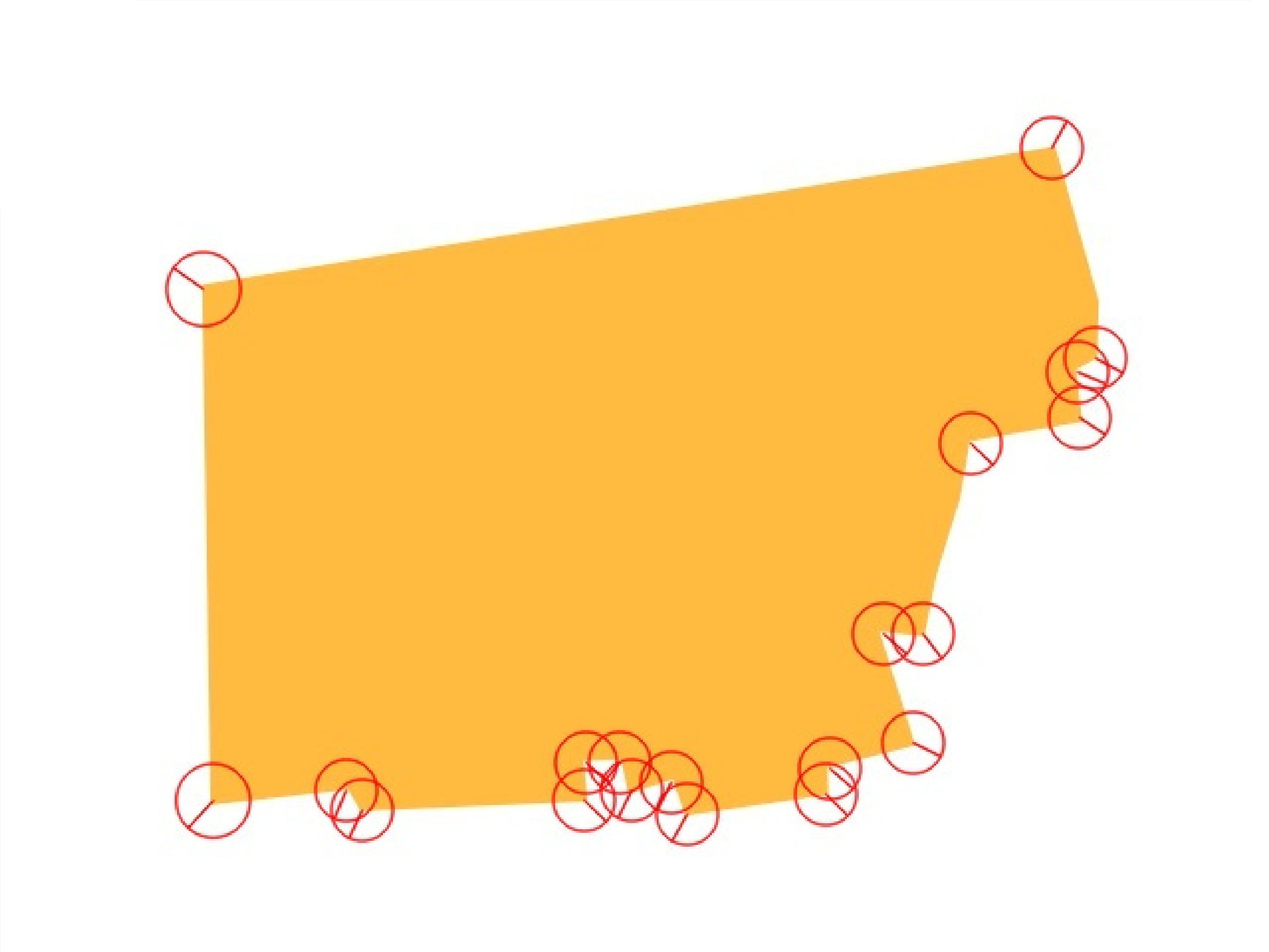




High complexity
10 to 20 corners

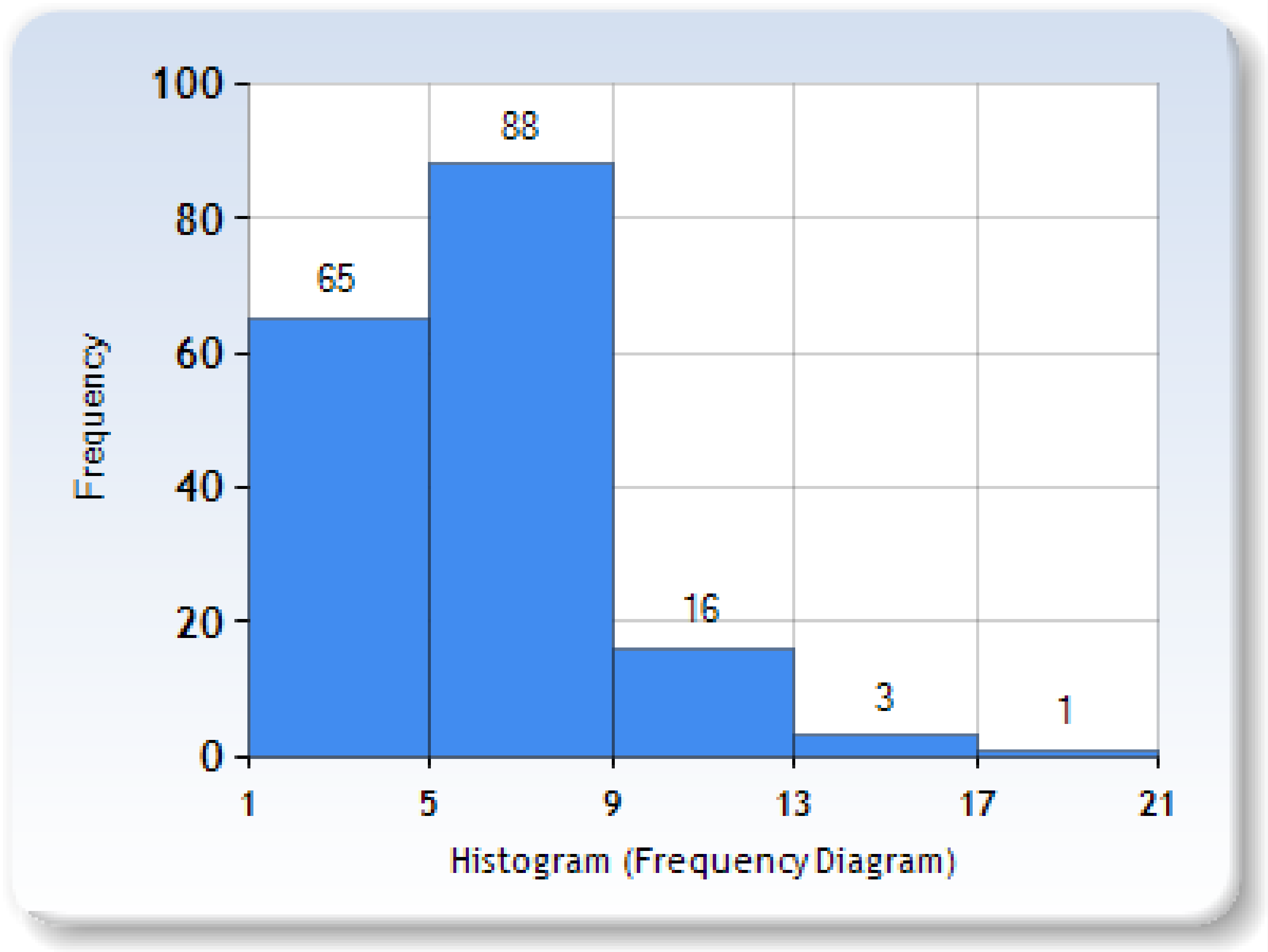






Depiction of number of layouts falling under each complexity class -

- Low complexity = 65
- Medium complexity = 88
- High complexity = 16+3+1 = 20



Your Histogram	
Mean	5.92486
Standard Deviation (s)	2.44477
Skewness	2.1652
Kurtosis	7.28835
Lowest Score	3
Highest Score	20
Distribution Range	17
Total Number of Scores	173
Number of Distinct Scores	13
Lowest Class Value	1
Highest Class Value	20
Number of Classes	5
Class Range	4

Frequency Table	
Class	*Count*
1-4	65
5-8	88
9-12	16
13-16	3
17-20	1

Retrieval of layouts

Model building

1. Data Preparation:

- length: Length of the tight-fitting box.
- breadth: Breadth of the tight-fitting box.
- empty_ratio: Ratio of the empty area to the tight-fitting box area.
- complexity: Number of sides of the building shape

2. Target Variable:
3. Family: Design family label (Each of 173 images are labelled)

3.Model Building Steps:

- In data preprocessing , we are already using ratios so. no need of normalization. Encode the family labels using label encoding or one-hot encoding.
- Feature Engineering: we are already taking enough features to determine the family labels.
- Out of all models k-Nearest Neighbors (k-NN) has been selected as it is :

1. Simple and easy to implement.
2. Can perform well with small datasets.
3. No training involved; it stores all the data points.

- Train the k-NN model on the entire dataset.
- When new input parameters are provided, use the trained k-NN model to find the k most similar designs from the dataset.

Other Suggestions

In the perspective of an architectural company,

1. The tight fitting box could be the dimensions of the plot (ground) that the customer owns
2. The layout (orange area) could be the actual area that the home will be designed of
3. The white area within the boundary box could tell us the way the home is designed - the structure of the surrounding area like the lawn area, portico etc.

Contd...

So once a customer approaches the architectural company, these three things, as mentioned before will be mentioned and we will be providing them with the layouts.

Additionally, we can also give them with floor plan ideas with the rooms.

Creating a new model

Hence we are planning to create a machine learning model that can give us the feasible room plan options, according to the given base layout, in a specific order.

NECESSITY	EXECUTION/IMPLEMENTATION
To leverage the knowledge of the architects Customers' limited knowledge on coming up with a sound floor plan	TRAINING DATA: PLAIN LAYOUTS AND FEASIBLE FLOORS PLANS WITH APPROPRIATE WEIGHTS TO THE ONES THAT ARE ARCHITECTURALLY SOUND SO ONCE THE MODEL IS TRAINED THE MODEL IS ACTUALLY EXPECTED TO GIVE THE DIFFERENT FLOOR PLANS WITH A RANKING OF THE MOST FEASIBLE AND SOUND DESIGN

Creating a reinforcement model

A reinforcement model that can use the given room designs to fit into the layouts and check if the area can be used optimally, using a threshold value

NECESSITY	EXECUTION/IMPLEMENTATION
The rooms are almost pre designed since they're the feasible options that the company has currently and they're mostly optimally designed in terms of area utilization and planning	USE A THRESHOLD VALUE TO THE LEFTOVER AREA OCCUPIED BY THE AVAILABLE OPTIONS OF ROOM DESIGNS IN THE COMPANY THE BASIC REQUIREMENTS OF THE ROOMS THAT WILL BE FIT IN THE LAYOUT WILL BE MET THE LAYOUTS ARE OPTIMISED ACCORDINGLY AND THE BEST LAYOUTS SURVIVE

Learnings and experiences

- Ideation and learning about different methods which could be used for grouping the images
- Learned about Hu moments and Siamese model
- Explored MobileNetV2
- Got to know about many features of open cv
- Explored image data for trends, and optimizations.
- Brainstormed over ideas and information which can be mined just by images
- Learned essential way of documentation of ideas

Hurdles faced.....

- DUPLICATE IMAGE-Before using the final approach ,we used other methods due to which many distinct images were considered as duplicate.After we used the final approach we got 173 distinct images.
- GROUPING OF IMAGES-While Edge detection ,the edges were not getting detected for some complex images.We tried many methods nd then finally wrote the correct code to extract the correct edges.
- Faced problem in detecting corners where edges are making more than 135 degree.
- Had to try on different models for answering Q3. posed to ensure the best method

THANK YOU