Scilab case study project on

Implementing Image Aesthetic Assessment via Drift Diffusion Model in Scilab

Debarupa Sinha

Jadavpur University

Image Processing, Distribution Prediction, Machine Learning

Date: 02-04-2026

Abstract

Image aesthetic assessment is inherently subjective. Traditional approaches predict single numerical scores or simple Gaussian distributions, hence failing to capture the diversity and complexity of human aesthetic perception. Different viewers may rate the same image quite differently, resulting in non-Gaussian score distributions. This case study, utilizing the Image Processing and Computer Vision Toolbox (IPCV) toolbox addresses the limitation by implementing a drift diffusion model inspired by psychologists that simulates the decision-making process of individuals. This model simulates psychological decision-making processes using positive and negative attractors to generate realistic distributions thus demonstrating the significance of integrating psychological theory in machine learning pipelines.

1) INTRODUCTION

1.1) Background

The last few decades have seen a rapid explosion in digital media, with that comes a strong demand for automatic evaluation of aesthetic assessment systems. Applications such as image retrieval, recommendation engines, photo enhancement, and content curation increasingly rely on computational models to evaluate the visual appeal of images. Traditional approaches to this problem have focused on classification to categorize images as **high** or **low** quality. However, these methods fail to capture the full distribution of human ratings, which can be bimodal or multi-modal depending on the image content and viewer diversity.

Early research in aesthetic assessment relied heavily on **handcrafted features**, inspired by principles from photography, art theory, and human visual perception. These features capture interpretable properties such as brightness, contrast, sharpness, color harmony, and compositional balance. Studies have shown that attributes like **image sharpness, entropy, edge density, and colorfulness** correlate strongly with perceived image quality. Although recent advances in deep learning—particularly convolutional neural networks (CNNs)—have significantly improved performance, handcrafted approaches remain valuable due to their **interpretability, lower computational cost, and suitability for environments with limited resources or tool constraints.**

1.2) Use of Drift Diffusion Model

This case study presents a Scilab-based pipeline that combines handcrafted image features extracted using the **IPCV toolbox** with a Drift Diffusion Model (DDM)-inspired deep distributional decomposition framework to predict aesthetic score distributions. **The AVA dataset** is used as the benchmark, where each image is associated with a distribution of human ratings. An exhaustive search procedure is employed to fit DDM parameters (m, n) using Earth Mover's Distance (**EMD**). A regression model is then trained to map image features to these parameters. The predicted parameters are used to reconstruct rating distributions, which are evaluated using KL divergence and EMD. The results demonstrate that interpretable handcrafted features, when combined with cognitive-inspired probabilistic modeling, can effectively

approximate human aesthetic judgments. A **Gaussian baseline comparison**, following the evaluation protocol of the reference paper (Tables 1 & 2) has also been added

1.3) Motivation of this case study

The motivation for this project is basically three fold:

- To provide a reference implementation of the drift diffusion model in an open-source environment.
- To showcase the IPCV toolbox as a powerful alternative to proprietary image processing solutions.
- To examine the utility of psychology driven decision making models for score distribution machine learning prediction tasks.

2) PROBLEM STATEMENT

In this case study, the problem is framed as a distribution prediction task over discrete rating bins (1–10), using the Aesthetic Visual Analysis (AVA) dataset as ground truth. Each image is associated with a normalized histogram of ratings provided by multiple human annotators. Instead of directly predicting this high-dimensional distribution, the approach leverages a Drift Diffusion Model (DDM)-inspired parameterization, where the distribution is generated through a stochastic evidence accumulation process governed by two parameters: the number of positive evidence steps (m) and negative evidence steps (n). The core problem, therefore, is to infer these latent parameters such that the simulated distribution closely matches the ground truth.

To solve this, the problem is broken into two sequential learning tasks:

- First– for each image, an **exhaustive search** is performed to estimate the optimal pair (m, n) that minimizes the difference between the simulated and ground truth rating distributions, using Earth Mover's Distance (EMD).
- Second– a supervised learning model is trained to learn the mapping from **IPCV extracted image features** to the corresponding parameters (m, n)

3) BASIC CONCEPTS RELATED TO THE TOPIC

a) Mechanism of Drift Diffusion Model (DDM):

First described by Ratcliff et al., 2016, DDM is an evidence accumulation model– in which with every trial, individuals accumulate evidence to base their response to a stimulus (in our case– images) until a threshold is reached. Each image provides some positive influences to the observer which improve their rating and negative influences which do the opposite along with random noise. The count of positive and negative influences are **m** and **n** respectively. The baseline score is the mid point 5. The final score is the baseline score plus the sum of positive influences and sum of negative influences + noise. The equations followed in this case study are

$$V = mid + \Sigma_i E_i^+ - \Sigma_j E_j^- + W$$

where:

- **mid** = baseline score (typically 5)
- **m** = number of positive attractors
- **n** = number of negative attractors
- $\mathbf{E_i}$ = evidence from each attractor (i varies from 1 to 6)
- **W** = random disturbance (uniformly distributed)

b) Stochastic Samplings:

Evidence Sampling (E) **&** Disturbance sampling (W) —

stochastic sampling refers to the generation of random evidence and noise components used to simulate human rating behavior

$$E = 0.5 * e^{-0.5*U(0,10)}$$

$$W = 0.015 * U(-1, 1),$$

NOTE : The equations are taken as it is from the referenced research article, though it was later observed an inverse transform method for evidence sampling also works quite well

c) Features Extracted:

c.1 Brightness Statistics

- Mean brightness reflects overall illumination
- Standard deviation captures contrast

c.2 Sharpness

Sharpness is computed using a **Laplacian filter**, which highlights high-frequency components such as edges. Images with higher sharpness are often perceived as clearer and more aesthetically pleasing.

c.3 Edge Density

Using a Sobel filter, edges are detected and thresholded. The proportion of edge pixels gives a measure of structural complexity.

c.4 Entropy

Entropy measures the information content or randomness of an image. Higher entropy often corresponds to richer textures and details.

c.5 Color Features

- Colorfulness captures variation in RGB channels
- Saturation reflects intensity of colors

Together, these features provide a compact representation of visual properties relevant to aesthetic perception.

d) Aesthetic Visual Analysis Dataset

The Aesthetic Visual Analysis (AVA) dataset is a large-scale benchmark for computational aesthetics research. Created by researchers at Yahoo Labs, it contains over 250,000 images collected from DPChallenge.com, a photography community website where users upload and rate photographs.

Due to the large size of the AVA dataset (approximately 33 GB), it was not feasible to download and process the complete dataset within the available computational and storage constraints. So,

a filtered subset of the dataset was used for experimentation. This subset is sufficient to demonstrate the effectiveness of the proposed pipeline.

e) Earth Mover's Distance

Earth Mover's Distance (EMD), also known as **Wasserstein-1 distance**, measures the minimum amount of work required to transform one distribution into another. For discrete distributions over scores 1-10, it can be computed as:

$$EMD(P,Q) = \sum_{i=1}^{N} |CDF_P(i) - CDF_Q(i)|$$

where CDF denotes the cumulative distribution function. EMD is symmetric and has nice properties: it considers the ordered nature of scores (unlike KL divergence), and it's more robust to small shifts in the distribution.

Interpretation:

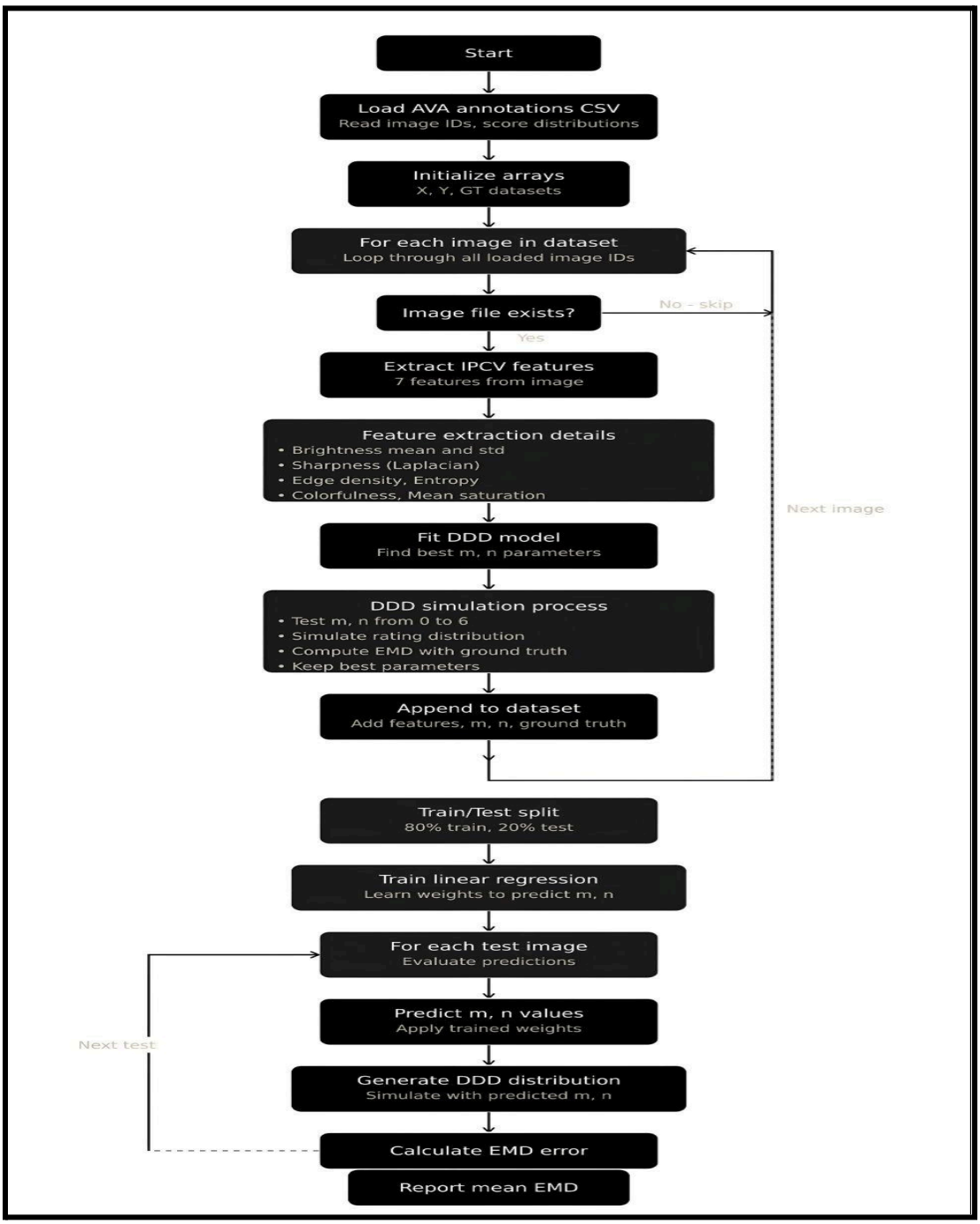
- EMD < 0.05: Excellent match
- 0.05 ≤ EMD < 0.15: Good match
- 0.15 ≤ EMD < 0.25: Moderate match
- EMD ≥ 0.25: Poor match

f) KL Divergence Metric

In information theory and statistics, Kullback-Leibler (KL) Divergence is a non-symmetric measure of the difference between two probability distributions. For two discrete probability distributions P and Q defined on the same probability space, the KL Divergence from Q to P is defined as:

$$D_{KL}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \log\left(\frac{P(x)}{Q(x)}\right)$$

4) FLOWCHART


Start
Load AVA annotations CSV
Read image IDs, score distributions
Initialize arrays
X, Y, GT datasets
For each image in dataset
Loop through all loaded image IDs
Image file exists?
No - skip
Yes
Extract IPCV features
7 features from image
Feature extraction details
• Brightness mean and std
• Sharpness (Laplacian)
• Edge density, Entropy
• Colorfulness, Mean saturation
Fit DDD model
Find best m, n parameters
DDD simulation process
• Test m, n from 0 to 6
• Simulate rating distribution
• Compute EMD with ground truth
• Keep best parameters
Append to dataset
Add features, m, n, ground truth
Next image
Train/Test split
80% train, 20% test
Train linear regression
Learn weights to predict m, n
For each test image
Evaluate predictions
Predict m, n values
Apply trained weights
Generate DDD distribution
Simulate with predicted m, n
Calculate EMD error
Report mean EMD
Next test

5. SOFTWARE / HARDWARE USED

- Operating System : Windows 10
- Scilab Version : 2026.0.1
- Toolboxes : IPCV 4.5.0.2
- Hardware : Intel Core i3 7th Generation, 4GB RAM

6. PROCEDURE OF EXECUTION

Setup and Installations:

1. Install Scilab
2. Install the IPCV toolbox : --> atomsInstall("IPCV")
3. Download the dataset which was used in the case study from here (or you can download the entire AVA dataset from Kaggle if you want to)
4. The folder has 1000 images with IDs and a CSV file containing the IDs with score distributions
5. Make sure the CSV files has 11 columns

Code Implementation details in steps:

Environment Setup: The workspace is cleared and the **IPCV** (Image Processing and Computer Vision) toolbox is loaded to handle image read and filter operations.

Annotation Loading: The script reads the filtered_aesthetic_scores.csv.

- It extracts the 10-bin rating frequency for each image.
- It **normalizes** these frequencies into a probability distribution (summing to 1).
- The **ground truth mean score** is calculated for each distribution.

Feature Engineering (IPCV):

Global Features: Mean brightness, standard deviation of brightness, and mean saturation are computed.

Structural Features: Sharpness is derived via a **Laplacian filter**, and edge density is calculated using a **Sobel operator**.

Complexity Scores: Image **Entropy** is calculated from a 32-bin histogram to measure information density.

DDM Parameter Optimization :

A grid search is performed over parameters m (positive drift steps) and n (negative drift steps).

For each (m,n) pair, the script simulates the Drift Diffusion process for 200 virtual raters.

The pair that results in the lowest **Earth Mover's Distance (EMD)** compared to the AVA ground truth is stored as the "optimal" label for that image.

Matrix Construction:

The extracted features are aggregated into a feature matrix X, and the optimized (m,n) pairs are stored in the label matrix Y.

Data Partitioning:

The dataset is split using an **80/20 ratio** into Training and Testing sets.

Linear Regression Training:

The script solves the Normal Equation to find the weight matrix. This maps the 7-dimensional visual feature vector to the 2-dimensional DDM parameter space.

Parameter Prediction:

For each image in the test set, the model predicts the values for m and n based on the image's visual features. These values are rounded and clamped to the original search range (0–6).

Distribution Synthesis:

The script runs the simulate_ddd function using the **predicted** parameters to generate a synthetic 10-bin aesthetic score distribution.

Baseline Comparison

For evaluation, a Gaussian baseline distribution is also constructed using:

- Mean (μ)
- Standard deviation (σ)

This Gaussian model acts as a reference probabilistic assumption and is compared against the DDM-generated distribution.

Performance Metric Calculation:

The performance of the model is evaluated using multiple distribution-level metrics. Earth Mover's Distance (EMD) is used as the primary metric for both parameter fitting and evaluation, as it captures the ordered nature of rating distributions. In addition, KL Divergence is computed to measure the divergence between predicted and ground truth distributions.

To establish a baseline for comparison, a Gaussian distribution is constructed for each image using its ground truth mean and standard deviation.The evaluation is structured in two parts: first, a bin-wise RMSE comparison between Gaussian and DDM predictions is reported, and second, statistical moment errors are computed for both models

7. Results

After running the script, the expected output in the console would show positive attractors (m) and negative attractors (n) predicted for each image along with their earth mover's distance (emd).

And Finally, the mean emd and KL metric is shown after evaluation on the test dataset completes. Expected Output is as follows :

```
"================================="
"FINAL RESULTS"
"DDM Mean EMD: 0.0639273"
"DDM Mean KL : 0.896516"
"================================="
```

Across the test samples, most EMD values lie in the range of approximately **0.02 to 0.07**, indicating that the predicted distributions are generally very close to the true distributions. A few outliers (e.g., around **0.16**) suggest occasional mismatches, likely due to cases where the underlying distribution is more complex or less well captured by the model.

It can be observed from the output that the predicted parameters (m,n) show a noticeable concentration around specific values (e.g., m=5, n=2 or 3).This clustering behavior reflects the limited expressive capacity of the linear regression model on the image features.

Side-by-side comparison of Ground Truth and Predicted Score Distributions of Sample Images

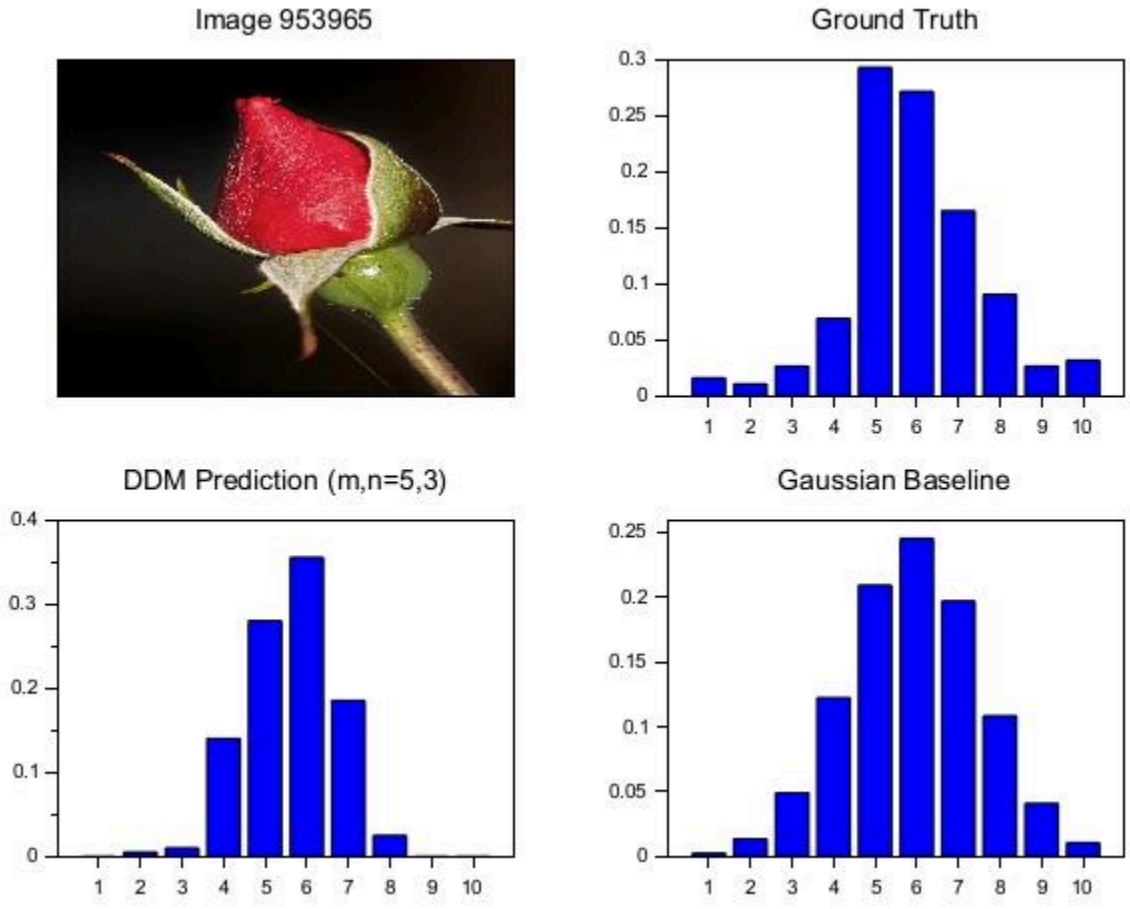
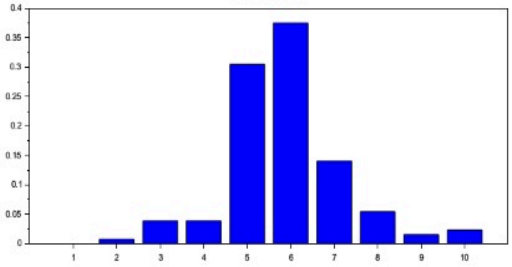
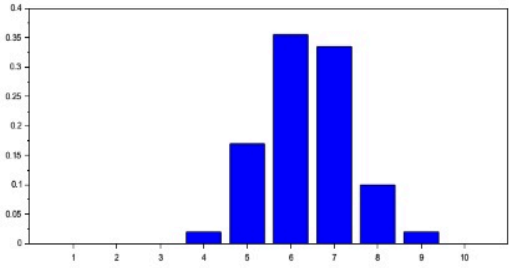


Image 953756

Ground Truth



DDM Prediction (m,n=6,1)



Gaussian Baseline

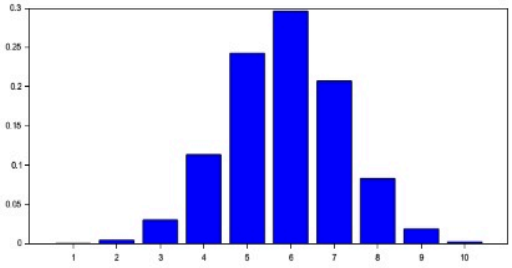
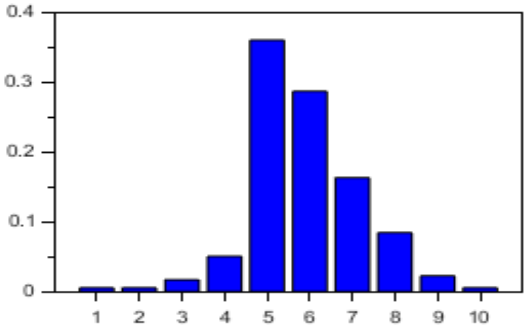
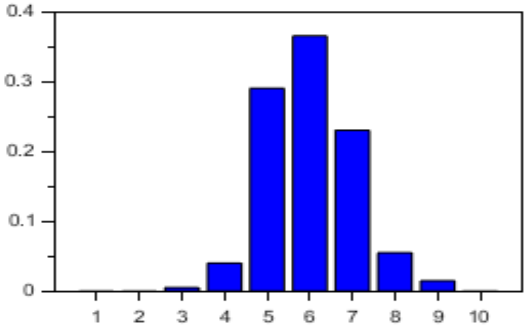


Image 953751

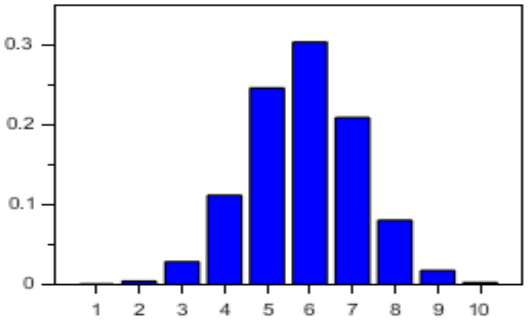
Ground Truth

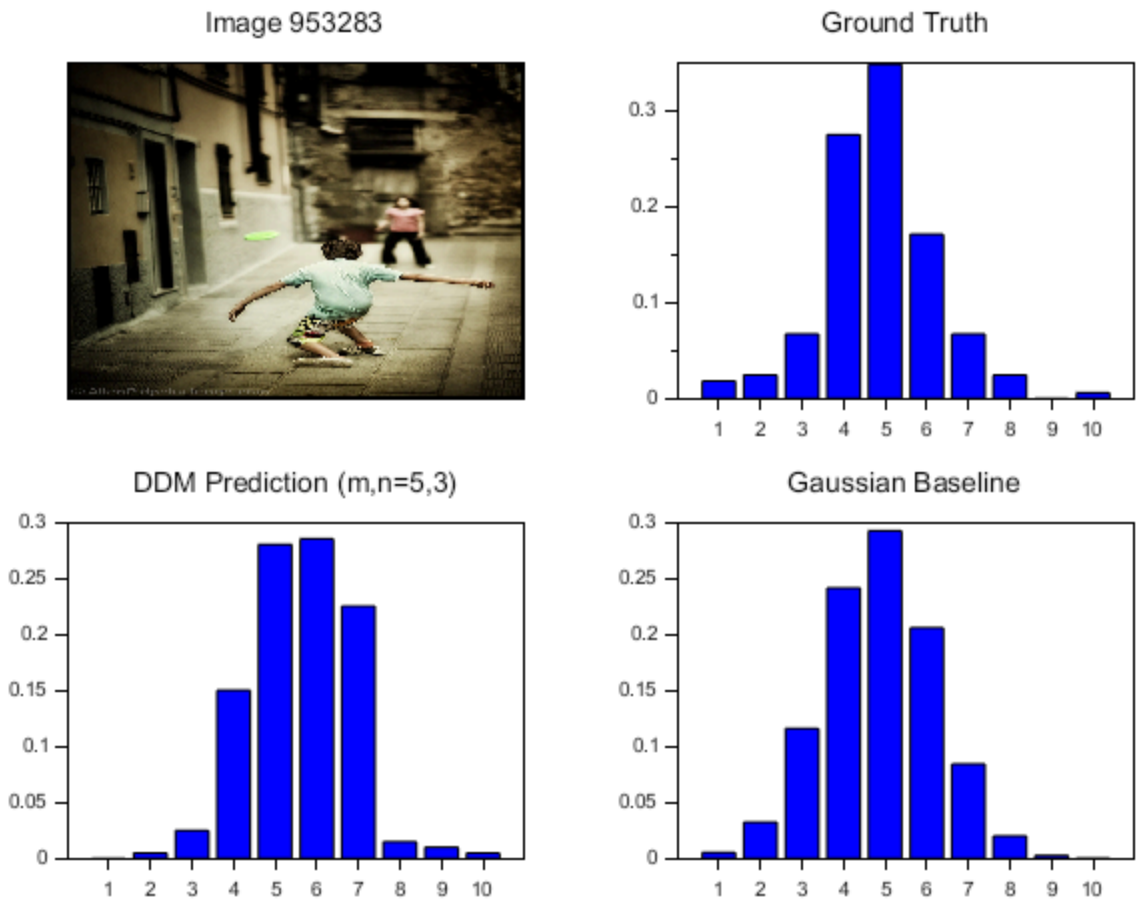


DDM Prediction (m,n=5,2)



Gaussian Baseline





Overall, the pipeline demonstrates that combining IPCV-based feature extraction with parameter estimation and a simple regression model can produce **accurate and stable approximations of aesthetic score distributions**. Further improvements could come from better feature representations or a more expressive model for predicting m, n.

Methods	**1-2**	**2-3**	**3-4**	**4-5**	**5-6**	**6-7**	**7-8**	**8-9**	**all**
Gaussian	**-**	**-**	**-**	**0.0300095**	**0.0376201**	**0.0316064**	**-**	**-**	**0.0347599**
DDM	**-**	**-**	**-**	**0.0730522**	**0.0509472**	**0.0627869**	**-**	**-**	**0.0589955**

Table 1: Fitting errors by Gaussian and Drift Diffusion Model using root mean square

Metric	Gaussian (MSE)	DDM Model (MSE)
Mean	0.0002201	0.2947672
Std Dev	0.0004796	0.1634005
Skewness	0.2712442	**0.1627259**
Kurtosis	2.2113344	**2.0596923**

Table 2. The fitting errors of four moments by Gaussian and Drift Diffusion Model, using Mean Square Error

Table 1 presents the Root Mean Square Error (RMSE) for both the Gaussian baseline and the Drift Diffusion Model (DDM) across specific aesthetic score bins. The recorded RMSE of 0 in the extreme ranges (1–4 and 7–9) is a direct result of data sparsity within the 1,000-image subset of the AVA dataset. This can be mitigated if a larger dataset is used. In case of Table 2, the Drift Diffusion Model (DDM) is a superior shape-predictor for skewness and kurtosis

Comparison with the referenced paper and disclosure of implemented and non implemented parts:

The numerical results are partially matching because the original paper uses a connected layer of GoogLeNet and ResNet model to extract deep features embeddings. In contrast, this Scilab implementation utilizes seven handcrafted features from the IPCV toolbox. While handcrafted features are computationally efficient, within the Scilab environment, they are quite limited to map visual cues to DDM parameters. The reference work employs deep neural networks to regress the parameters m and n from the image features. This case study utilizes a Linear Regression model to map the IPCV features to the DDM parameters. The core logic of the Drift-Diffusion Model (DDM) has been successfully replicated. Like the reference paper, this implementation moves beyond single-score prediction to model the full aesthetic score distribution (1–10) and uses Earth Mover's distance and KL Divergence as evaluation metrics. The project serves as a working proof-of-concept showing that Scilab, along with its image processing computer vision toolbox , can validate the utility of DDM theory for image aesthetic prediction.

Future works and scope of improvements:

- More advanced Feature Representations can be employed. Incorporating deep features such as CNN embeddings could significantly improve performance.
- Computational constraints limit the dataset, a more diverse dataset utilising the entire aesthetic visual analysis dataset or integrating with other similar datasets can enhance model robustness to unseen samples.
- Replacing Regression with more expressive architectures such as multi layer perceptrons and support vector regression can help mapping the non linear features with the parameters more effectively.
- Future iterations of this project can leverage Scilab's neural networks toolbox to replace the linear regression mapping
- The pipeline can be extended to analyse moving images or video frames to study how the drift parameters can vary over a sequence of frames

8) REFERENCES

1. Xin Jin, Xiqiao Li, Heng Huang, Xiaodong Li, Xinghui Zhou, “A Deep Drift-Diffusion Model for Image Aesthetic Score Distribution Prediction”, 2020 arXiv:2010.07661.
2. Myers CE, Interian A and Moustafa AA (2022) A practical introduction to using the drift diffusion model of decision-making in cognitive psychology, neuroscience, and health sciences. Front. Psychol. 13:1039172. doi: 10.3389/fpsyg.2022.1039172
3. IPCV github
4. Dataset Used: ava