IMAGE DATING MODEL

Sidhant Jain, Mukund Arun, Mihir Rao

PROBLEM STATEMENT

How can we estimate the approximate year in which a digital photograph was captured using only the visual information contained within the image?

APPLICATIONS

- ***Digital Archive Verification***

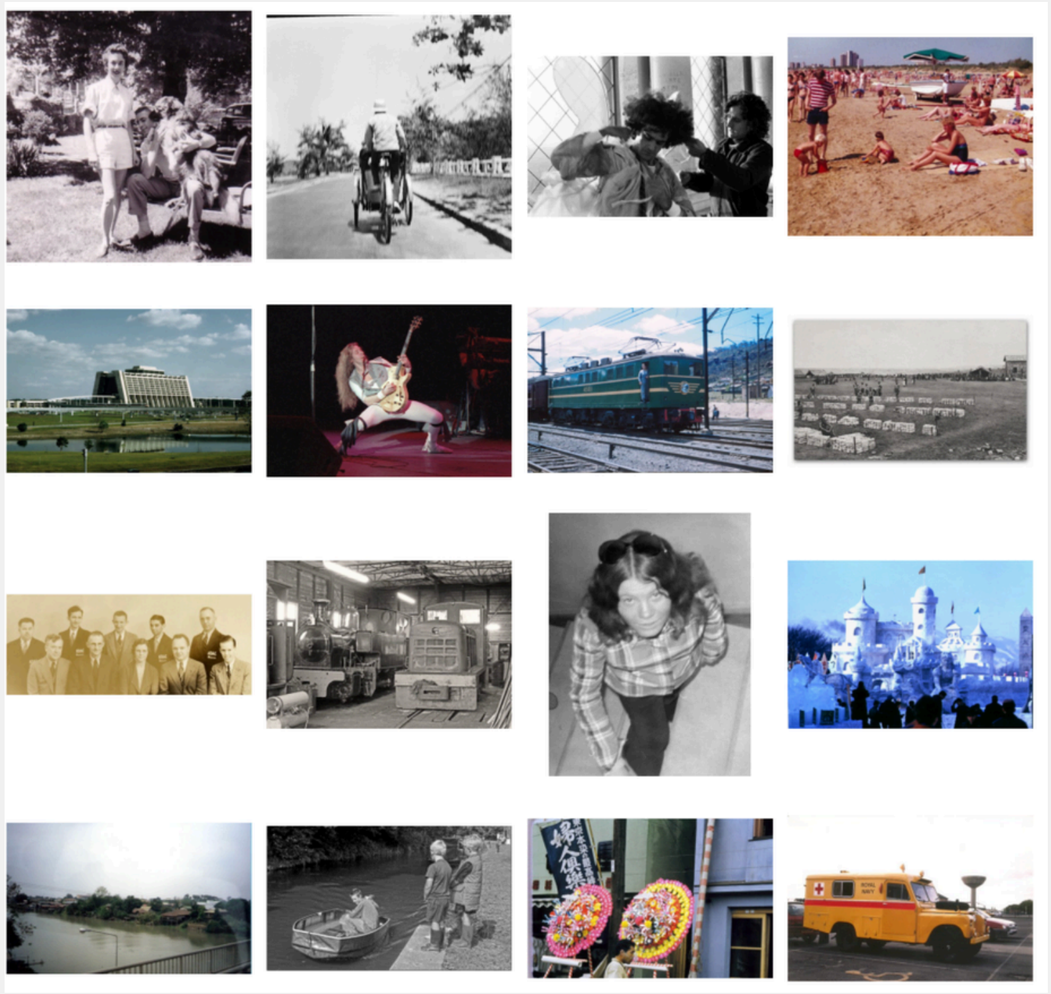
Large photo archives contain millions of images with missing or approximate timestamps. Our model could act as a verification layer to refine dates and organize collections. (Example: Library of Congress)

- ***Media Authenticity Verification***

Can help identify inconsistencies between the claimed date of images circulating in news or online media and the visual cues present in the photograph.

- ***Consumer Photo Dating Service***

Can estimate the approximate time period of old family photographs



LITERATURE REVIEW

Paper 1: “Dating Historical Color Images” (ECCV 2012)

- Used handcrafted statistical features such as hue histograms and RGB co-occurrence patterns with a Linear SVM classifier to predict the decade of historical colour photographs (1930s–1970s).
  - Linear SVM classifiers were trained for decade-wise classification
  - *Achieved an accuracy of 45.7%*

Limitations:

- *Dataset was just around 1400 images.(1930-1970)*
- *Only looked at coloured images*

Source

*Palermo, F., Hays, J., Efros, A.A. (2012). Dating Historical Color Images. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds) Computer Vision – ECCV 2012. ECCV 2012

https://graphics.cs.cmu.edu/projects/historicalColor/

LITERATURE REVIEW

Paper 2: Temporal Image Forensic Analysis for Picture Dating with Deep Learning (Farah ahmed et al,2020)

- *Used deep CNN models (AlexNet and GoogLeNet) to estimate when digital photos were taken.*
- *Trained on a new, time-sequenced database (NTIF)*
- *Achieved 80-88% accuracy*

Limitations:

- *The dataset only spans approximately 2 years (2014-2016), limiting the model's ability to learn broader long-term temporal variations.*
- *The approach requires images from the same camera device, limiting generalization across different devices*

Source

https://researchportal.northumbria.ac.uk/ws/portalfiles/portal/177416971/FarahAhmedThesisUNN

LITERATURE REVIEW

Paper 3: “When Was This Picture Taken? - Image Date Estimation in the Wild” (ECCV 2014)

- Used pre trained CNN (GoogleNet) as a feature extractor
- The extracted CNN features are used as input to a regression model that predicts the capture year
- Achieved 43% accuracy

Limitations

- The approach relies heavily on deep CNN features, leading to higher computational cost and lower explainability compared to handcrafted feature-based methods.
- CNN-based feature extraction increases processing time compared to simpler feature-based methods.

Source

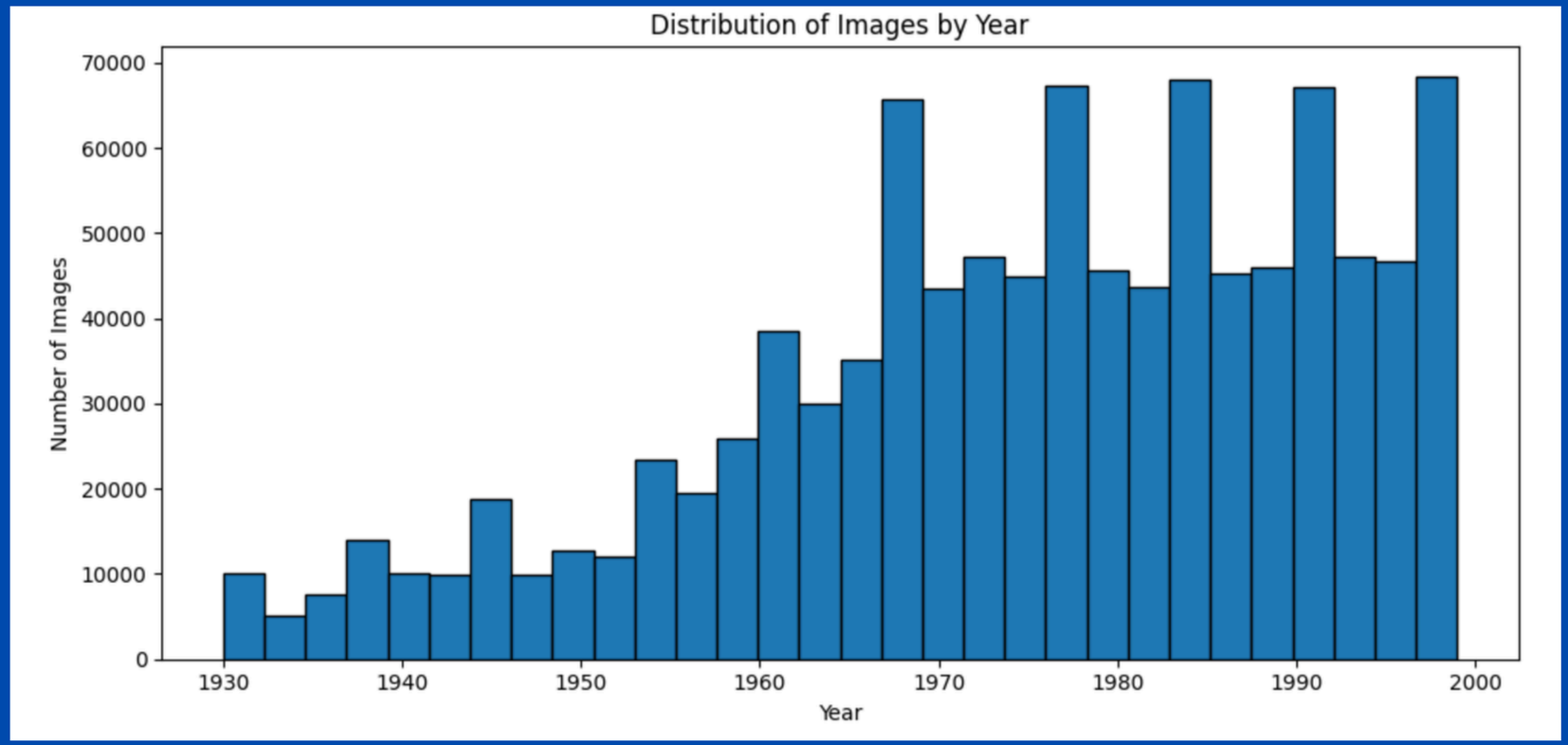
https://link.springer.com/chapter/10.1007/978-3-319-56608-5_57#Sec4

DATASET

The dataset consists of 1.029,710 Flickr images between 1930 to 1999.

Each row in the dataset has two fields: the year in which the image was taken and the url to the image.

For balanced training, 250 images per year were selected due to lower image availability in certain years



[1] Eric Müller, Matthias Springstein, and Ralph Ewerth: "When Was This Picture Taken?" - Image Date Estimation in the Wild.
In: Advances in Information Retrieval: Proceedings of 39th European Conference on Information Retrieval (ECIR), Aberdeen (UK), Lecture Notes on Computer Science (LNCS), Vol. 10193, Springer, pp. 619-625, 2017.

FEATURE PREPROCESSING

```
Processed image 0
Processed image 1
Processed image 2
Skipping image 3: cannot identify image file <_io.BytesIO object at 0x793be440d4e0>
Processed image 4
Skipping image 5: cannot identify image file <_io.BytesIO object at 0x793b9db5dd50>
Processed image 6
Processed image 7
Processed image 8
Processed image 9
Processed image 10
Processed image 11
Processed image 12
Processed image 13
Skipping image 14: cannot identify image file <_io.BytesIO object at 0x793b9db5dd50>
Processed image 15
Processed image 16
Skipping image 17: cannot identify image file <_io.BytesIO object at 0x793b9db5dd50>
Processed image 18
Skipping image 19: cannot identify image file <_io.BytesIO object at 0x793b947815d0>
```

Choice of numerical features:

- statistical intensity features
- RGB color statistics
- brightness and contrast measures
- shape descriptors, edge and gradient features
- texture variance
- inter-channel color differences
- RGB histogram features

	mean_red	mean_green	mean_blue	saturation_mean	brightness_mean
0	0.224451	0.447759	0.438167	-1.671476	0.376691
1	0.676698	0.861024	0.545769	0.455461	0.778766
2	-0.810598	-0.880866	-0.954691	-1.070257	-0.881009
3	-0.483152	-0.325586	-0.022675	0.239071	-0.355002
4	0.042222	-0.056151	-1.113010	1.027201	-0.139481

	brightness_std	laplacian_variance	noise_variance	edge_density
0	0.434849	-0.635698	0.292790	-0.554703
1	0.141853	-0.411994	0.127119	0.317754
2	-0.467379	2.849437	-2.498294	2.143514
3	-0.538508	-0.721300	1.059801	-0.861447
4	-0.962636	-0.700314	0.888013	0.065701

	image_entropy	url
0	0.259831	https://farm8.staticflickr.com/7563/1605875696...
1	0.557131	https://farm9.staticflickr.com/8189/8418118218...
2	0.626994	https://farm4.staticflickr.com/3331/4644893892...
3	0.510998	https://farm9.staticflickr.com/8355/8418136138...
4	-0.234581	https://farm6.staticflickr.com/5513/9078605425...

Missing Data occurring due to the URL not loading was skipped.

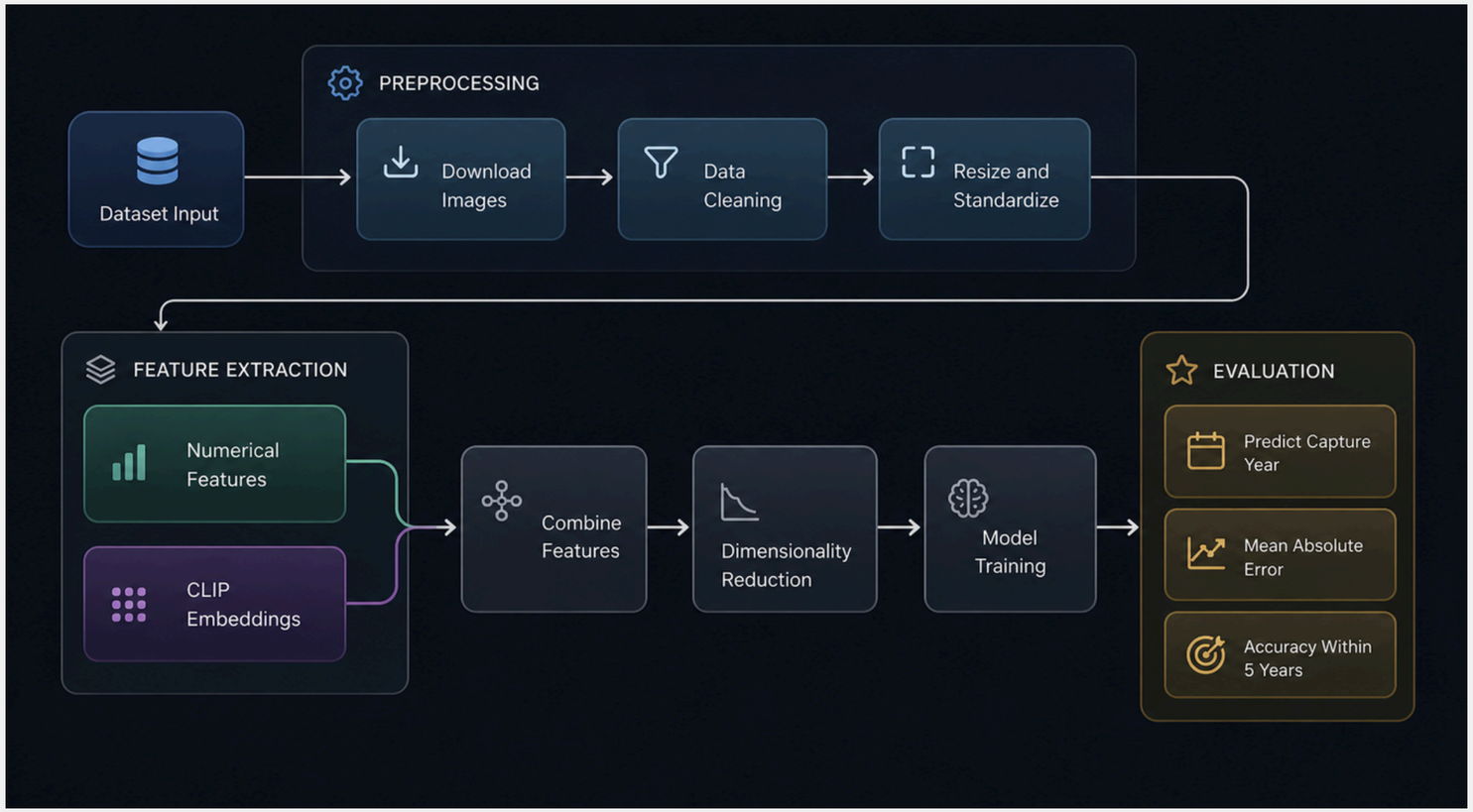
FEATURE PREPROCESSING

	0	1	2	3	4	5	6	7	8	9	...	502	5
0	-0.008753	-0.003167	-0.001196	0.010965	0.023902	-0.016530	0.003883	0.040873	-0.058092	0.021162	...	-0.020983	0.0395
1	-0.031580	0.017768	0.013866	0.002712	-0.066248	-0.024927	0.024680	-0.005183	0.001112	0.035509	...	-0.011126	-0.0233
2	-0.030213	0.028634	-0.012537	0.013425	0.005911	-0.019133	0.040308	0.003683	0.008286	0.015892	...	-0.055206	-0.0027
3	-0.049860	0.003221	0.034161	0.018473	-0.015277	-0.026999	-0.017124	-0.016942	0.031706	0.007295	...	-0.010895	-0.0026
4	-0.026387	0.021258	-0.009597	0.006186	-0.003807	-0.001918	-0.048136	-0.021182	-0.040450	-0.024710	...	-0.017757	-0.0108

5 rows × 512 columns

- Extracted **512-dim image embeddings** using CLIP
- Goal of the Hybrid Model: combine low-level statistics with high-level semantics for better predictions
- Made use of a CNN (YOLO v8), but ended up discarding it
- Applied PCA to reduce CLIP embedding dimensionality from 512 to 256 features.

METHODOLOGY

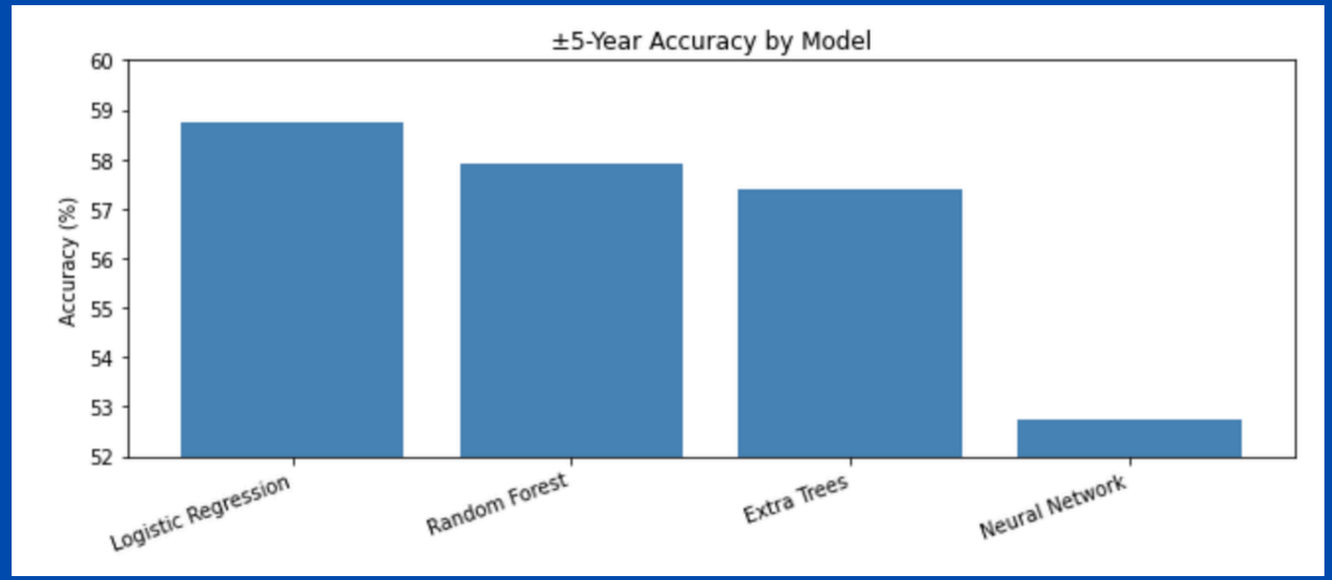


THE PROCESS

- Data collection & preprocessing
- Image resizing & standardization
- Numerical feature + CLIP embedding extraction
- Feature fusion & dimensionality reduction
- Model training & year prediction
- Evaluation using MAE and ±5 year accuracy

PERFORMANCE METRICS

- Achieved 58.74% ±5-year accuracy and an MAE of 7.31 years using Logistic Regression
- Benchmarked against Random Forest (57.95%), Extra Trees (57.38%), and a neural network baseline (50.63%, MAE 8.18 years)



POSSIBLE CHALLENGES FOR SOLUTION

Challenge	Why it matters
Increased computational cost for generating CLIP embeddings	CLIP inference is expensive and becomes a major bottleneck at scale.
Real world archive collections are temporally skewed - war decades and cultural events	The model was trained on a balanced 250-per-year sample, so it has no prior for how images are actually distributed in deployment
GPU dependency for efficient deployment	Storing high-dimensional vectors for millions of images becomes substantial.
Training window is hard-capped at 1930-1999 with no out-of-distribution detection	A consumer submitting a photo from 1920 or 2003 gets a confidently wrong answer with no warning

THANK YOU