Online Appendix for *Improving Computer Vision Interpretability: Transparent Two-level Classification for Complex Scenes*

Stefan Scholz[1], Nils B. Weidmann[1], Zachary C. Steinert-Threlkeld[2], Eda Keremoğlu[1], and Bastian Goldlücke[1]

[1]Center for Image Analysis in the Social Sciences, University of Konstanz
[2]Luskin School of Public Affairs, University of California, Los Angeles

May 29, 2024

A Protest Image Dataset

The Protest Image Dataset is a new data collection project that includes images from social media with an emphasis on political protests. The dataset contains, besides the images themselves, variables on location, time, and a hand-annotated protest variable. This section describes the general conventions guiding the image collection and image annotation.

A.1 Image Collection

Selection of Countries We used the Armed Conflict, Location, and Event Dataset (ACLED) to analyze all protests since January 1, 2014 to the present (Raleigh et al., 2010). We then identified the twenty country-years with the most protest events for each of ACLED's 16 regions, resulting in 313 candidate country-years[1]. Next, the logistic regression model from Steinert-Threlkeld et al. (2022) was used to identify the 46 countries (171 country-years) with enough people and income to produce enough protest images from Twitter. These 46 were narrowed to 14 based on their Polity IV score and region, with a goal of generating broad coverage of regime types and parts of the world. With 14 countries we could ensure that we annotate a sufficient number of images per country despite the restriction due to the labor-intensive annotation of images.

Selection of Posts Because of its widespread use throughout the world (Huang & Carley, 2019), we used the social media platform Twitter to obtain protest images posted by observers on the ground. In order to assign a tweet to a specific country, we required that the tweet was geolocated within this country. Though it is possible that users who geolocate their tweets are not a representative sample of their country's population (Malik et al., 2021),

[1]ACLED did not start covering North America until Mexico in 2018; its coverage guide lists Mexico as Central America, but in the dataset the country is coded as North America. The United States was added in 2020, Bermuda, Canada, and Saint Pierre and Miquelon in 2021. Every candidate country-year for North America was therefore included at this stage of the selection.

work comparing Twitter users who share protest images to those who share non-protest images finds no differences between those two groups (Steinert-Threlkeld et al., 2022). As the storage requirements would render it impossible for us to collect within a whole year all geolocated tweets, particularly in the large countries, we decided to narrow down the tweets for each country to a specific date range. Thus, we continued to analyze the 14 countries' number of protests per month, and chose a date range that includes both the rise and the fall of the protests. For most countries, we specified the start date on the first of the rising month and the end date on the last of the falling month. We made an exception for countries where we expected a particularly large number of tweets; we specified their start and end dates also within the courses of these months. For all countries, we ensured that within these date ranges, in addition to tweets posted during high numbers of protests, we also included tweets posted seven days before and after the protest period. After selecting these periods, we extracted tweets from the relevant country-days from a corpus of tweets downloaded from Twitter's POST statuses/filter endpoint. This extraction resulted in just over 135 million tweets which were then used to find protest images.

Selection of Images Twitter allows a tweet to have multiple media; it allows up to four photos, one animated GIF or one video. In our selection of images, we included all media that Twitter categorized as a photo, but no media was categorized as a GIF or video. We then downloaded these images and saved them together with their tweet identifier, tweet date and media identifier. Despite our previous selection of tweets, in some countries we collected far too many images to store them in the space available to us, not to mention annotate them in the next step. Therefore we decided to introduce a limit of images per country at 100,000. This limit affected 11 countries, where to stay below the limit we randomly sorted the images and then downloaded them until we had 100,000. For example, in Japan, the country where we collected the most images, 5,546,059 images were sampled to 100,000. In contrast, the Kazakhstan tweets contained only 52,825 images, so we kept all of them.

A.2 De-Duplication

We analyze the occurrence of duplicates in our dataset to rule out possible problems: Through many duplicates of the same image, the importances of certain features could be inflated. In addition, if the same image occurs in the training set as well as in the testing set, the classification results would be biased. We generate encodings for the images by propagating them through a convolutional neural network. We use a MobileNet v3 (Howard et al., 2017) pretrained on the ImageNet dataset (Deng et al., 2009) and sliced at the last layer. This generates an encoding of 576 features. We compute the cosine similarity between all pairs of images and retrieve duplicates with a similarity equal or larger than 0.99. We then identify 127,769 duplicate images in 48,905 clusters. The duplicate images are dropped from the dataset.

	Region	Country	Date range	Images
1	Northern Africa	Algeria	2019-02-01 – 2020-03-01	100,000
2	Middle East	Lebanon	2019-10-01 – 2020-01-15	42,203
3	Middle East	Bahrain	2016-01-01 – 2017-12-31	100,000
4	South America	Argentina	2020-05-01 – 2020-09-30	100,000
5	South America	Chile	2019-10-01 – 2019-12-31	100,000
6	South America	Venezuela	2019-01-01 – 2019-11-30	100,000
7	Eastern Africa	Ethiopia	2015-11-01 – 2016-12-31	5,867
8	Southern Africa	South Africa	2021-01-01 – 2021-08-31	100,000
9	Western Africa	Nigeria	2018-09-01 – 2019-09-30	100,000
10	Caucasus and Central Asia	Kazakhstan	2019-01-01 – 2020-03-30	52,825
11	Europe	Russia	2019-07-07 – 2019-10-06	100,000
12	Southeast Asia	Indonesia	2019-05-01 – 2019-10-31	100,000
13	East Asia	Japan	2018-02-22 – 2018-06-30	100,000
14	Southeast Asia	Philippines	2017-05-01 – 2017-12-31	100,000

Table A1: Selection of images

A.3 Image Annotation

We annotate images as to whether they display a political protest, or part of it. We define protest as

- **A publicly visible event or action**: It takes place in a public space and therefore can be observed by the public.
- **An event involving one or more participants that are present on site**: Protest can range from individual statements to mass demonstrations. We exclude instances where a symbol or an item is displayed publicly without the presence of protesters themselves.
- **A political statement or expression**: An objection or a criticism against a political actor or institution. This can be achieved by means of anything not corresponding to the norm and thus attracting public attention; it can be done by verbal statements or speeches, but also with banners or symbols.

Images often cover protests only partially; for example, they display a single person or a group of persons participating in the protest. These images are considered "protest" images, if their relation to a protest as defined above can be ascertained. They do not need to display a complete protest event. The coder's annotation is coded on a four-point scale as

- **Protest (high certainty)**: The coder is certain that the image shows a part of a protest as defined above.
- **Protest (low certainty)**: The coder believes that the image probably shows a part of a protest as defined above.

- **No protest (low certainty)**: The coder believes that the image probably does not show a part of a protest as defined above. statements or speeches, but also with banners or symbols.
- **No protest (high certainty)**: The coder is certain that the image does not show a part of a protest as defined above.

We present the coders in the first round with 6,000 images from each country. These images are randomly selected from the previously selected images. In the second, third and fourth round, we select from each country 3,000 images by weighted random sampling. To calculate the weights, we train a model on the already annotated images. This model is based on a vision transformer (ViT, Dosovitskiy et al., 2020); it is retrained after each round of annotations. This model gives us for every not-yet-annotated image a score between 0 and 1, where a low score indicates a likely-non-protest image and a high score a likely-protest image. The images are then grouped by these scores in 20 equal-width bins, and their weights are calculated such that the probability of drawing an image from one bin is the same as from another bin. The aim of this weighted random sampling is to reduce the probability of likely-no-protest images and increase the probability of likely-protest images. The annotation of the images proceeds until the coders' available time is used up.

A.4 Analyzing Reliability Across Coders

We analyze the degree that coders consistently assigned categorical protest ratings to the images in our dataset. The protest annotations for 141,538 images were done by four coders. For 65,120 images we have annotations from two coders.

Cohen's kappa was computed for four classes (no protest high, no protest low, protest low, protest high), with an inter-rater reliability of 0.68. When we combine high and low confidence ratings to obtain a binary classification, we obtain an inter-rater reliability of 0.81. According to McHugh (2012), these results indicate moderate and strong reliability, respectively. Since each image is annotated by a random set of coders, we decide to also compute the intraclass correlation correlation coefficient (ICC1). This is equal to a one-way ANOVA fixed effects model. For four classes (no protest high, no protest low, protest low, protest high) this gives us a intraclass correlation of 0.83. For two classes (no protest, protest) this gives us a intraclass correlation of 0.79, which is a good result according to Koo and Li (2016).

A.5 Splitting Images into Training and Testing Set

Table A2 presents the number of images per country in the four annotation categories. These annotated images in the dataset are randomly split into a training and testing set. The training set contains 80% of the images, whereas the testing set contains 20%.

	No Protest		Protest	
	High	Low	Low	High
Argentina	12,864	124	343	1,087
Bahrain	14,070	156	111	172
Chile	9,708	445	1,016	2,776
Algeria	9,756	214	617	2,450
Indonesia	13,845	267	215	434
Lebanon	11,082	279	586	2,128
Nigeria	13,565	222	177	309
Russia	14,040	113	135	300
Venezuela	11,006	322	656	1,790
South Africa	13,815	78	94	171
Total	123,751	2,220	3,950	11,617

Table A2: Images in protest images dataset annotated in different protest categories and countries.

B Training of Models

For the models we train using our segment-based approach, we choose four different classification methods: logistic regression, simple decision trees, collections of decision trees and gradient-boosted decision trees. We use logistic regression because it is widely used by social scientists, and to provide a benchmark against. We choose the tree-based models because they intuitively allow us to vary the complexity and interpretability of the models. As implementation of the collections of decision trees a random forest (Breiman, 2001) is used, for gradient-boosted decision trees XGBoost (Chen & Guestrin, 2016) is used.

In order to make a comparison with conventional computer vision methods, we also have to make a selection of these methods. We decide to train a convolutional neural network (CNN) by ourselves. We decide to train a ResNet50 because we want to keep the training times and hardware requirements lower compared to, for instance, a ResNet101 or Resnet152 (He et al., 2016). It also allows a direct comparison to the same architecture but trained on a different dataset by Won et al. (2017). In addition, we select a vision transformer (ViT) because they have shown to outperform CNNs on many computer vision tasks while requiring less computational resources (Dosovitskiy et al., 2020). These vision transformers are available as base, large and huge-sized variants. We make sure to use a base-sized variant of the ViT to make the comparison to the Resnet50 as fair as possible. Our ViT model refers to a base-sized variant with a patch resolution of 16x16 and a fine-tuning resolution of 384x384.

The first step in the training of each segment model is to select the hyperparameters. For this purpose, a 5-fold cross-validation is performed for the complete grid of hyperparameters. For the logistic regression, different regularization strengths are tried, with up to 10 improving the accuracy. For the simple decision trees, the maximum depth is varied from 1 to 16. From a depth of 8 to 16, most classifiers improve only minimally, or even deteriorate. For the random forests, the number of trees, the number of maximum features, the maximum depth and the minimum number of samples in a leaf are varied. The number of trees is varied from 1 to 1,000, with more trees leading to no obvious improvement. For the gradient-boosted trees, a large number of hyperparameters is varied, the maximum depth, the number of boosting rounds, learning rate, and minimum loss reduction. If we disable boosting (number of boosting rounds 0), the maximum F1 score is achieved with a maximum depth of 8. The score deteriorates if the maximum depth is above or below 8. In order to look at the effect of the number of boosting rounds, we fix the maximum depth at 8. The F1 scores improve with more boosting rounds, until 10,000 boosting rounds.

For the training of the ResNet50 and ViT, we use pre-trained weights on the ImageNet dataset. This way, the model knows from the beginning certain features that are independent of our protest images, such as corners, edges, shapes, etc. We never use the trained weights of the ResNet50 by Won et al. (2017), also not as pre-trained weights for our self-trained ResNet50. During the training of the models, however, these pre-trained weights could be completely changed, as no layers are frozen and the gradients for all weights in all layers are calculated and changed. For the sake of readability, we have decided to use the term training. But by the definition of finetuning, this "training" procedure could also be

referred to as "finetuning" procedure. We decide to use a cross entropy loss with a stochastic gradient descent optimizer with momentum. For hyperparameter tuning, the training data is additionally split into a training (80%) and validation set (20%). This is not the same as the 5-fold cross validation for the segments models, but fulfills a similar purpose with significantly less computational effort. We follow best practice for setting most of the hyperparameters. But we optimize the values for the learning rate and momentum with the help of hyperparameter tuning. It is found that a learning rate of 1e-03 is best for the ResNet50, while it is significantly lower for the ViT at 1e-05. The optimal momentum is found to be 0.99 for the ResNet50 and 0.99 for the ViT. After the optimal hyperparameters for the models are found, they are retrained on the entire training data for 100 epochs.

All models are trained on a server node with 8 Intel Xeon @ 2.50 GHz cores, 128 GB memory as well as a NVIDIA graphics card, Quadro RTX 6000 with 24 GB memory. From the segment models, the logistic regression and gradient-boosted trees need the longest training time – but not more than 12 minutes. The final model of the ResNet50 is trained in 15 hours, while the ViT is trained in 30 hours. To infer whether the images in our dataset are protest images, the gradient-boosted tree needs 2 minutes (0.0007 seconds/image). The inference with the ResNet50 needs 6 minutes (0.0025 seconds/image) and the ViT 30 minutes (0.0129 seconds/image).

C Main Results

We provide the full results for the full combination of different design choices in Table A3. Results for the conventional image classification methods are provided at the bottom of the table.

	Training			Testing		
	Precision	Recall	F1	Precision	Recall	F1
Segments (COCO, bin, logistic)	0.6495	0.3118	0.4213	0.6249	0.3013	0.4066
Segments (COCO, count, logistic)	0.6527	0.4621	0.5411	0.6589	0.4729	0.5506
Segments (COCO, area max, logistic)	0.1691	0.0028	0.0055	0.1667	0.0026	0.0051
Segments (COCO, area sum, logistic)	0.2289	0.0067	0.0131	0.2135	0.0061	0.0119
Segments (COCO, bin, tree)	0.7672	0.3713	0.5004	0.5735	0.2856	0.3813
Segments (COCO, count, tree)	0.5225	0.7304	0.6092	0.5187	0.7180	0.6023
Segments (COCO, area max, tree)	0.7629	0.5931	0.6674	0.5163	0.4019	0.4520
Segments (COCO, area sum, tree)	0.7642	0.5741	0.6557	0.5376	0.4022	0.4601
Segments (COCO, bin, forest)	0.8011	0.2658	0.3991	0.6762	0.2120	0.3228
Segments (COCO, count, forest)	0.8082	0.5625	0.6633	0.6984	0.4767	0.5666
Segments (COCO, area max, forest)	0.6486	0.3249	0.4329	0.5527	0.2830	0.3743
Segments (COCO, area sum, forest)	0.8400	0.3608	0.5047	0.6808	0.2679	0.3845
Segments (COCO, bin, boosted tree)	0.7362	0.4567	0.5637	0.6237	0.3701	0.4645
Segments (COCO, count, boosted tree)	0.7091	0.5860	0.6417	0.6836	0.5699	0.6216
Segments (COCO, area max, boosted tree)	0.9156	0.7140	0.8023	0.6374	0.4330	0.5157
Segments (COCO, area sum, boosted tree)	0.8129	0.5734	0.6724	0.6486	0.4305	0.5175
Segments (LVIS, bin, logistic)	0.7509	0.5831	0.6565	0.7389	0.5818	0.6510
Segments (LVIS, count, logistic)	0.7254	0.5138	0.6016	0.7048	0.5130	0.5938
Segments (LVIS, area max, logistic)	0.4443	0.0480	0.0867	0.4259	0.0443	0.0803
Segments (LVIS, area sum, logistic)	0.5374	0.1137	0.1877	0.5521	0.1124	0.1868
Segments (LVIS, bin, tree)	0.8976	0.8005	0.8463	0.5942	0.5429	0.5674
Segments (LVIS, count, tree)	0.8831	0.7778	0.8271	0.6476	0.5667	0.6044
Segments (LVIS, area max, tree)	0.7650	0.5149	0.6155	0.7154	0.4626	0.5618
Segments (LVIS, area sum, tree)	0.9219	0.8364	0.8771	0.6097	0.5596	0.5836
Segments (LVIS, bin, forest)	0.9568	0.5087	0.6642	0.8415	0.3736	0.5175
Segments (LVIS, count, forest)	0.9592	0.5719	0.7165	0.8256	0.4333	0.5684
Segments (LVIS, area max, forest)	0.9431	0.5416	0.6881	0.7817	0.3762	0.5079
Segments (LVIS, area sum, forest)	0.9761	0.5455	0.6999	0.8457	0.3784	0.5229
Segments (LVIS, bin, boosted tree)	0.9944	0.9753	0.9848	0.7594	0.6315	0.6896
Segments (LVIS, count, boosted tree)	0.9982	0.9813	0.9897	0.7834	0.6624	0.7178
Segments (LVIS, area max, boosted tree)	1.0000	1.0000	1.0000	0.7805	0.6569	0.7134
Segments (LVIS, area sum, boosted tree)	**1.0000**	**1.0000**	**1.0000**	**0.7821**	**0.6675**	**0.7203**
ResNet50 (Won et al., 2017)	0.5834	0.4698	0.5205	0.5787	0.4584	0.5116
ResNet50 (self-trained)	0.8508	0.8192	0.8347	0.7657	0.7327	0.7489
ViT (self-trained)	**0.9199**	**0.8939**	**0.9067**	**0.8400**	**0.8060**	**0.8226**

Table A3: Evaluation of different methods. "Self-trained" means trained on the images collected for this project.

To compare the performance between countries, we present results for the best conventional method (vision transformer model, ViT) and the best of our two-level classifiers (LVIS vocabulary, area sum features and a boosted tree classifier). Table A4 presents the results for the images in the 10 countries of our dataset.

		Training			Testing		
		Precision	Recall	F1	Precision	Recall	F1
Segments	Argentina	1.0000	1.0000	1.0000	0.7500	0.6167	0.6769
	Bahrain	1.0000	1.0000	1.0000	0.3803	0.4821	0.4252
	Chile	1.0000	1.0000	1.0000	0.8560	0.5884	0.6974
	Algeria	1.0000	1.0000	1.0000	0.8933	0.7651	0.8243
	Indonesia	1.0000	1.0000	1.0000	0.5490	0.6462	0.5936
	Lebanon	1.0000	1.0000	1.0000	0.8646	0.7053	0.7769
	Nigeria	1.0000	1.0000	1.0000	0.5463	0.6082	0.5756
	Russia	1.0000	1.0000	1.0000	0.5488	0.5172	0.5325
	Venezuela	1.0000	1.0000	1.0000	0.7804	0.7342	0.7566
	South Africa	1.0000	1.0000	1.0000	0.5000	0.5472	0.5225
ViT	Argentina	0.9019	0.8924	0.8971	0.8284	0.7735	0.8000
	Bahrain	0.8423	0.8238	0.8330	0.6667	0.6786	0.6726
	Chile	0.9196	0.8708	0.8945	0.8569	0.7586	0.8048
	Algeria	0.9552	0.9560	0.9556	0.9103	0.9103	0.9103
	Indonesia	0.8755	0.7996	0.8359	0.7731	0.7077	0.7390
	Lebanon	0.9187	0.9056	0.9121	0.8396	0.8287	0.8341
	Nigeria	0.8834	0.7789	0.8279	0.7065	0.6701	0.6878
	Russia	0.9217	0.8793	0.9000	0.7674	0.7586	0.7630
	Venezuela	0.9225	0.9121	0.9173	0.8343	0.8446	0.8394
	South Africa	0.8274	0.7689	0.7971	0.6000	0.5660	0.5825

Table A4: Evaluation of different methods per country. The Vision Transformer (ViT) is the best conventional method, whereas Segments is the best of our two-level classifiers with the LVIS vocabulary, area sum features and a boosted tree classifier.

D Analysis of Clustered Images

We analyze the performance of our classifier on subcategories of protest images. To identify these subcategories, we use an unsupervised approach that clusters the images and thus assigns them to unlabeled categories (see Zhang and Peng (2022)).

In order to do this, we extract an embedding for each image in our dataset. This embedding is generated by our self-trained vision transformer (ViT) in the last linear layer, and is 768 features long. We then cluster the embeddings using the Euclidean distance and the KMeans algorithm. To determine the number of clusters, the number of clusters is raised as long as the coherence of each cluster is given. This is done according to the procedure proposed by Zhang and Peng (2022) by always selecting 20 random images from each cluster, determining a topic for that cluster, and checking if at least 50% of the images in that cluster match the topic. This procedure leads to 30 clusters.

Then, we evaluate the accuracy separately on each cluster. As classifier we use our best two-level classifier (LVIS vocabulary, area sum features and a boosted tree classifier). We do not analyze the accuracy on the training images, as these images are all correctly classified and are therefore also correctly classified in the individual clusters. Instead, the accuracy on the test images is analyzed based on the true negatives, false negatives, true positives, false positives, precision score, recall score and F1 score in the individual clusters.

Table A5 shows the accuracy of the classifier in the clusters that contain at least 20 protest images from the test set. In the cluster of protest images with flags, the precision and recall score are close to each other, which indicates that the classifier is balanced to make errors in classifying either as a protest image and a non-protest image. In the other clusters, however, the precision is higher than the recall. This indicates that in these clusters the classifier makes more errors in classifying non-protest images as protest images than protest images as non-protest images. At the same time, this shows that the accuracy differs between the clusters. The differences in the precision and recall scores considered above are also reflected in the F1 scores. By comparing the F1 scores of the clusters, we see that an F1 score of 0.4156 is achieved for the clusters with African gatherings, with fire smoke of 0.5039, state police of 0.5221 and gatherings of 0.6222. This means that for these clusters it is below the F1 score of 0.7203, which the classifier achieves on all test images. In contrast, it achieves a higher accuracy for the clusters with large mass protests with an F1 score of 0.7495, protests with signboards with 0.8137 and protests with flags with 0.9370. The less accurate clusters can possibly be explained by the fact that object categories are missing for them in the LVIS vocabulary. For example, the vocabulary contains no categories related to fire, smoke and policemen. The difficulties with gatherings could be explained by the fact that it is difficult to distinguish whether it is a protest image or a non-protest image based on the number of people. But especially if there are large masses on the images, the classifier has a good performance, also if objects and flags can be seen on the protest images.

In addition to the clusters shown in Table A5, there are also clusters that contain less than 20 protest images. These include, for example, a cluster with football matches in which the classifier correctly classifies 406 non-protest images, misclassifies 1 protest image as non-protest image and incorrectly classifies 7 non-protest images. In this case, the F1 score is not

		TN	FN	TP	FP	Precision	Recall	F1
9	Edited images	1,268	31	17	17	0.5000	0.3542	0.4146
10	Streets	1,161	55	41	39	0.5125	0.4271	0.4659
14	Protest with flags	14	58	714	38	0.9495	0.9249	0.9370
16	Gathering	414	137	210	118	0.6402	0.6052	0.6222
17	Fire smoke	242	101	64	25	0.7191	0.3879	0.5039
18	African gatherings	504	19	16	26	0.3810	0.4571	0.4156
22	Large mass protests	108	167	365	77	0.8258	0.6861	0.7495
23	State police	288	74	59	34	0.6344	0.4436	0.5221
24	Protest with signboards	120	174	450	32	0.9336	0.7212	0.8137
27	Flags	287	88	94	46	0.6714	0.5165	0.5839
30	Random images with text	1,884	67	31	44	0.4133	0.3163	0.3584

Table A5: Evaluation of best two-level classifier for clusters that contain at least 20 protest images from the test set. Evaluation metrics are true negatives (TN), false negatives (FN), true positives (TP), false positives (FP), precision score, recall score and F1 score.

defined because there are no true positive cases, which is why precision and recall are zero and the F1 score is not defined. Also in the cluster of concert images 450 non-protest images are classified correctly, 10 non-protest images are classified incorrectly, 7 protest images are classified incorrectly, 1 protest image is classified correctly. This leads to a precision score of 0.0909, a recall score of 0.1250 and an F1 score of 0.1053. These low scores can also be explained by the small number of protest images in this cluster.

To get a visual impression of the clusters, we select sample images from the clusters. To get a representative impression of the clusters, the images are selected according to the centrality in the cluster. For this purpose, the distances of the images to the cluster centroid are calculated in each cluster. Images whose distances are lower are more central in the cluster, whereas images whose distances are higher are further outside the cluster.

Figure A1 shows three images for each of the clusters containing at least 20 protest images in the test set. The left images are drawn from the first tercile, the middle images from the second tercile and the right images from the third tercile of each cluster.

Figure A1: Sample images for clusters than contain at least 20 protest images from the test set. The left, middle and right images are drawn from the first, second and third terciles of the distances to the cluster centroids.

E Results using Secondary Dataset

The paper's primary dataset uses high and low certainty protest images as protest images and high and low certainty non-protest images as non-protest images. This coarsening may introduce noise, so we repeat the analysis with a secondary dataset using only the high confidence protest and non-protest images.

Table A6 shows the fit statistics for the resulting models. All model fits improve and the rank ordering does not change. Figure A2 shows the object categories occupying the largest areas of protest images and the important objects of protest images. Importance results are largely the same, though chairs take kites' place and cars drop out for hats. Figure A3 shows the variation of object importance by country. The results for posters, cars and candles are the same.

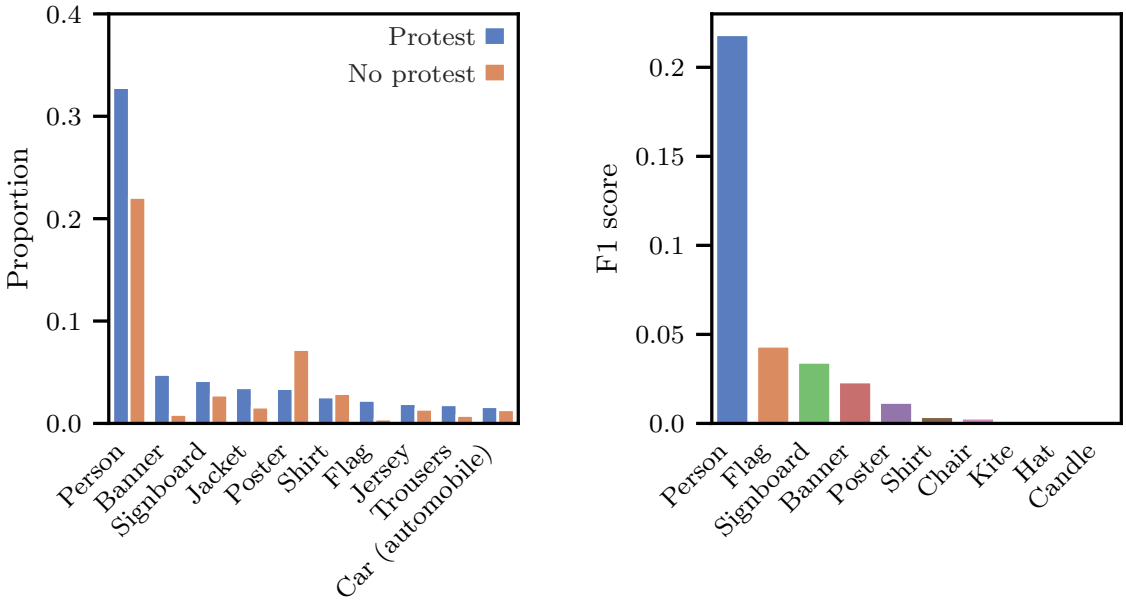

Figure A2: Proportion of segments on high confidence protest and non-protest images (left) and importance of area-sum aggregated segments (right) on images that have been annotated with high confidence.

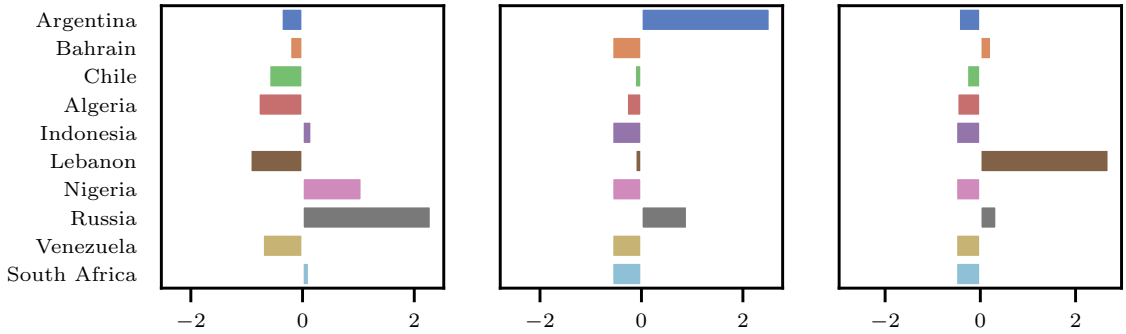

Figure A3: Differences in importance in different countries of posters (left), cars (middle), and candles (right) on images that have been annotated with high confidence.

	Training			Testing		
	Precision	Recall	F1	Precision	Recall	F1
Segments (COCO, bin, logistic)	0.6622	0.3143	0.4263	0.6309	0.2986	0.4054
Segments (COCO, count, logistic)	0.6555	0.4712	0.5483	0.6607	0.4768	0.5539
Segments (COCO, area max, logistic)	0.1241	0.0018	0.0036	0.0690	0.0009	0.0017
Segments (COCO, area sum, logistic)	0.1629	0.0039	0.0076	0.1698	0.0039	0.0076
Segments (COCO, bin, tree)	0.7797	0.3930	0.5226	0.5602	0.2883	0.3807
Segments (COCO, count, tree)	0.5233	0.7011	0.5992	0.5296	0.7009	0.6033
Segments (COCO, area max, tree)	0.7881	0.5979	0.6799	0.5129	0.3941	0.4457
Segments (COCO, area sum, tree)	0.7707	0.6184	0.6862	0.5235	0.4225	0.4676
Segments (COCO, bin, forest)	0.6382	0.2766	0.3859	0.5707	0.2517	0.3494
Segments (COCO, count, forest)	0.8051	0.5486	0.6525	0.7105	0.4901	0.5801
Segments (COCO, area max, forest)	0.8525	0.3496	0.4959	0.6754	0.2435	0.3580
Segments (COCO, area sum, forest)	0.8685	0.3759	0.5247	0.6752	0.2612	0.3767
Segments (COCO, bin, boosted tree)	0.7558	0.4305	0.5486	0.6507	0.3559	0.4601
Segments (COCO, count, boosted tree)	0.7439	0.6084	0.6694	0.6999	0.5800	0.6344
Segments (COCO, area max, boosted tree)	0.9373	0.7413	0.8279	0.6504	0.4290	0.5170
Segments (COCO, area sum, boosted tree)	0.7827	0.5154	0.6216	0.6700	0.4281	0.5224
Segments (LVIS, bin, logistic)	0.7813	0.6290	0.6969	0.7705	0.6183	0.6861
Segments (LVIS, count, logistic)	0.7526	0.5402	0.6290	0.7312	0.5301	0.6146
Segments (LVIS, area max, logistic)	0.4146	0.0423	0.0768	0.4017	0.0396	0.0721
Segments (LVIS, area sum, logistic)	0.5257	0.1014	0.1700	0.5588	0.1063	0.1786
Segments (LVIS, bin, tree)	0.7628	0.5247	0.6217	0.7119	0.4806	0.5739
Segments (LVIS, count, tree)	0.7906	0.5621	0.6571	0.7477	0.5151	0.6099
Segments (LVIS, area max, tree)	0.7880	0.5809	0.6688	0.7228	0.5262	0.6091
Segments (LVIS, area sum, tree)	0.7865	0.5819	0.6689	0.7208	0.5275	0.6092
Segments (LVIS, bin, forest)	0.9257	0.4746	0.6274	0.8321	0.3881	0.5293
Segments (LVIS, count, forest)	0.9488	0.5926	0.7295	0.8113	0.4514	0.5800
Segments (LVIS, area max, forest)	0.9681	0.5842	0.7287	0.8265	0.4139	0.5516
Segments (LVIS, area sum, forest)	0.9878	0.5815	0.7321	0.8874	0.4002	0.5516
Segments (LVIS, bin, boosted tree)	0.9977	0.9877	0.9927	0.7950	0.6639	0.7236
Segments (LVIS, count, boosted tree)	0.9906	0.9550	0.9725	0.8124	0.6876	0.7448
Segments (LVIS, area max, boosted tree)	1.0000	1.0000	1.0000	0.8325	0.6863	0.7524
Segments (LVIS, area sum, boosted tree)	**1.0000**	**1.0000**	**1.0000**	**0.8301**	**0.7001**	**0.7596**
ResNet50 (Won et al., 2017)	0.5686	0.5305	0.5489	0.5694	0.5245	0.5460
ResNet50 (self-trained)	0.8987	0.8428	0.8698	0.8251	0.7612	0.7919
ViT (self-trained)	**0.9516**	**0.9271**	**0.9392**	**0.8917**	**0.8649**	**0.8781**

Table A6: Evaluation of different methods on images that have been annotated with high confidence. "Self-trained" means trained on the images collected for this project.

F Temporal Analysis

We analyze how the prevalence of the segments changes over the course of a protest. Having collected the images in our dataset based on protest periods in countries, we can track their prevalence before, during and after the protests. To do this, we use the LVIS segments that we detected on the images in our dataset. From these segments, we sum up the occurrence of segments for each country, day and segment type.

Figure A4 displays the frequency of the segments over time. In each country, it is limited to the three most frequent segments that are detected in that country over the entire period.

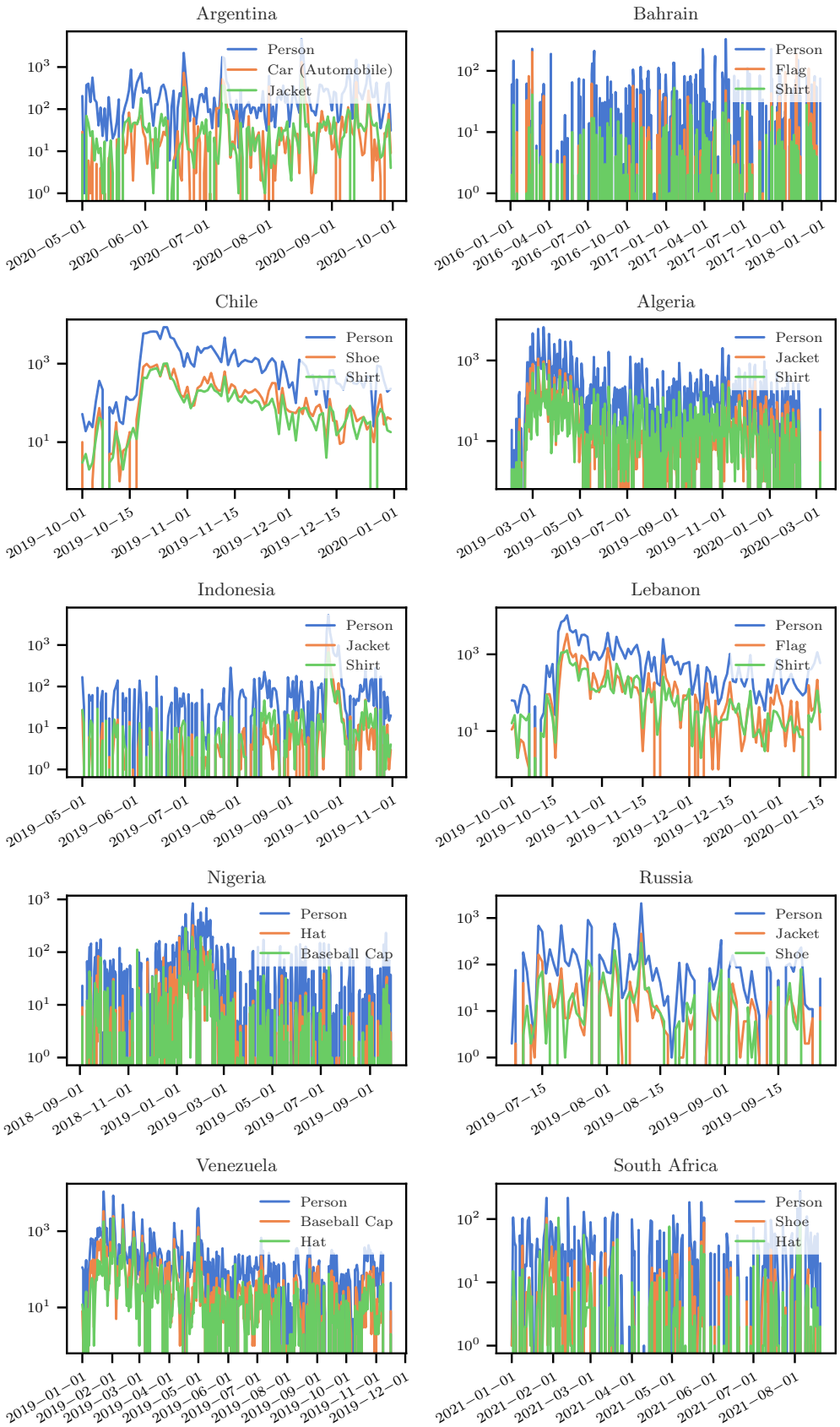

Figure A4: The three most common segments per country and their frequency over time.

G Analysis of Segments

Our main analysis in the paper shows which object categories are identified by a machine to be important for recognizing a protest image. However, are these categories also considered to be important by humans? To find out, we conduct an additional validation exercise at the level of segments (not entire images), asking a coder which segments they deem important for recognizing a protest image.

For this task, we create a subsample of our protest image dataset consisting of 100 random protest images (high or low confidence) from each of the 10 countries. These images are inspected by one of our coders, who then have to complete two coding steps. In the first step, the coder is asked to look at the protest image and name up to three objects that the coder considers most important to identify it as a protest image. The identification of these objects is done on the raw images. In the second step, after the objects have been freely named, the same protest image is shown with the segments highlighted. The segments shown are those from the LVIS vocabulary (Gupta et al., 2019) that were recognized by the segmentation model by Zhou et al. (2022), with a confidence score of at least 0.1 (as for the analysis in the paper). Importantly, the segment categories are not shown for these segments, they are simply numbered. The coder is then asked whether the objects identified as important in the first step correspond to one of the segments shown. Because some of the images contain a large number of segments, making it difficult to find the correct identifiers, the coding tool is configured such that the coder could interactively click through the segments to find the right segments and numbers.

The coder identifies 2,210 objects as important on the 1,000 protest images. The ten most frequent object names (freely chosen by the coder) are: people (776), flag (430), poster (216), signboard (195), banner (118), mask (97), police (84), person (69), fire (43) and kid (15). These categories largely overlap with those from our two-stage classification method, which identifies people as the most important objects, followed by flags, signboards, banners and posters. Our method does not identify police officers, fires and children because they are not included in the LVIS vocabulary as separate categories. This shows that custom adaptations of the segmenting method for specific tasks will likely improve results, as we discuss in the paper. A first result is that at a general level, the object categories identified by our two-stage method largely match those that coders consider relevant for the identification of protest images.

We also test whether *all* segments identified by the coder could be matched to LVIS segments. For the vast majority, this is possible. It is only for 141 objects (6.4%) that there is no corresponding segment. These include categories that are included in the LVIS vocabulary (for example, 24 posters, 22 persons, 19 flags), but which the segmenter fails to identify on the respective images. For the successfully assigned objects we check whether the objects indeed match the segments. To do this, we compare the object names given by the coder with the ones detected by the segmenter. We set up a small dictionary to ensure that object names that are spelled slightly differently could be recognized as identical. For a strict matching (object names identical), we find that 1,512 out of the 2,210 segments (68%) are correctly detected by the segmenter. For a lenient matching (the dictionary

incorporates subtypes and supertypes as well), the number of correctly detected segments increases to 1,626 (74%). Objects that are repeatedly incorrectly recognized are objects that are interpreted as weapons by our coder. This analysis shows that human and machine largely rely on the same segments to code protest images.

H Collecting Images in the Future

The data for this paper were collected in real time using R's `streamR` package (Barbera, 2018). One of the authors maintained a continuous connection to Twitter's filtered stream endpoint and requested only tweets with location information. Collecting tweets agnostically in real-time means any event of sufficient magnitude is collected automatically, obviating the need for researchers' to search for events *post hoc*. Searching for events after they occur also risks introducing sample bias, as searches rely on keywords or specifying users and content could be deleted between its posting and the researcher's download.

The past tense is used in the previous paragraph because Elon Musk's purchase of Twitter has led to severely restricted data access. The free tier only returns 1,500 tweets per month. The Basic tier for $100/month provides only 10,000 tweets. The Pro, $5,000 and 1 million. At the Pro level, one can stream tweets, but they count against the 1 million quota, which would be reached in less than a day without stringent filter rules. Access equivalent to what this paper had requires the Enterprise tier. That pricing is available upon request in contrast to the $0 price before Musk's neutering. Except for three alternatives or academics with corporation-level resources, Twitter access is over.

The three alternatives are using already downloaded tweets, scraping, and the European Union's Digital Services Act (DSA). If one has previously downloaded tweets, it is possible and easy to download media from those tweets. Each tweet contains a `image_url` field, and access to those media are not rate limited. While old tweets may no longer be available (Pfeffer et al., 2023), if they are then their images are. As of April 2022, *hiQ v. LinkedIn* and then *Van Buren v. United States*, decided at the United States' Ninth Circuit Court of Appeals and the Surpreme Court, respectively, establish that the Computer Fraud and Abuse Act does not allow companies to prevent scraping of their public data. A researcher can build a scraper themselves or use the Python package `snscape`. The other option is to apply for research access as permitted pursuant to Article 40 of the DSA. Doing so requires an application. As of this writing, we are not aware of any Twitter research conducted as a result of the DSA.

This paper's method is applicable to any image data, however, not just those from Twitter. Other sources of images include Facebook pages, Instagram, Telegram, and WhatsApp; Pexels, Tumblr, and Unsplash; news archives; or stills of videos from TikTok and YouTube. A golden age of online social media data appears to have ended, but researcher creativity will ensure that research does not.

References

Barbera, P. (2018, December). streamR: Access to Twitter Streaming API via R.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794.

Deng, J., Dong, W., Socher, R., Li, L.-J., Kai Li, & Li Fei-Fei. (2009). ImageNet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Gupta, A., Dollar, P., & Girshick, R. (2019). LVIS: A dataset for large vocabulary instance segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Huang, B., & Carley, K. M. (2019). A large-scale empirical study of geotagging behavior on Twitter. *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*, 365–373.

Koo, T. K., & Li, M. Y. (2016). A guideline of selecting and reporting intraclass correlation coefficients for reliability research. *Journal of Chiropractic Medicine*, *15*(2), 155–163.

Malik, M., Lamba, H., Nakos, C., & Pfeffer, J. (2021). Population bias in geotagged tweets. *Proceedings of the International AAAI Conference on Web and Social Media*, *9*(4), 18–27.

McHugh, M. L. (2012). Interrater reliability: The kappa statistic. *Biochemia Medica*, 276–282.

Pfeffer, J., Mooseder, A., Lasser, J., Hammer, L., Stritzel, O., & Garcia, D. (2023). This Sample Seems to Be Good Enough! Assessing Coverage and Temporal Reliability of Twitter's Academic API. *Proceedings of the International AAAI Conference on Web and Social Media*, *17*, 720–729.

Raleigh, C., Linke, A., Hegre, H., & Karlsen, J. (2010). Introducing ACLED: An armed conflict location and event dataset: Special data feature. *Journal of Peace Research*, *47*(5), 651–660.

Steinert-Threlkeld, Z. C., Chan, A. M., & Joo, J. (2022). How state and protester violence affect protest dynamics. *The Journal of Politics*, *84*(2), 798–813.

Won, D., Steinert-Threlkeld, Z. C., & Joo, J. (2017). Protest activity detection and perceived violence estimation from social media images. *Proceedings of the 25th ACM International Conference on Multimedia*, 786–794.

Zhang, H., & Peng, Y. (2022). Image clustering: An unsupervised approach to categorize visual data in social science research. *Sociological Methods & Research*, 004912412210826.

Zhou, X., Girdhar, R., Joulin, A., Krähenbühl, P., & Misra, I. (2022). Detecting twenty-thousand classes using image-level supervision. *European Conference on Computer Vision*, 350–368.