NoisyArt: A Dataset for Webly-supervised Artwork Recognition

R. Del Chiaro, Andrew Bagdanov and A. Del Bimbo
Media Integration and Communication Center, University of Florence, Italy

Keywords: Cultural Heritage, Computer Vision, Instance Recognition, Image Categorization, Webly-supervised Learning.

Abstract: This paper describes the NoisyArt dataset, a dataset designed to support research on webly-supervised recognition of artworks. The dataset consists of more than 90,000 images and in more than 3,000 webly-supervised classes, and a subset of 200 classes with verified test images. Candidate artworks are identified using publicly available metadata repositories, and images are automatically acquired using Google Image and Flickr search. Document embeddings are also provided for short descriptions of all artworks. NoisyArt is designed to support research on webly-supervised artwork instance recognition, zero-shot learning, and other approaches to visual recognition of cultural heritage objects. Baseline experimental results are given using pretrained Convolutional Neural Network (CNN) features and a shallow classifier architecture. Experiments are also performed using a variety of techniques for identifying and mitigating label noise in webly-supervised training data.

1 INTRODUCTION

Cultural patrimony and exploitation of its artifacts is an extremely important economic driver internationally. This is especially true for culturally dense regions like Europe and Asia who rely on cultural tourism for jobs and important industry. U.S. tourist travelers alone represent nearly 130 million adults spending approximately $171 billion annually on leisure travel (Chen and Rahman, 2018). Museums are massive, distributed repositories of physical and digital artifacts. For decades now museums have been frantically digitizing their collections in an effort to render their content more available to the general public. Initiatives like EUROPEANA (Valtysson, 2012) and the European Year of Cultural Heritage[1] have advanced the state-of-the-art in cultural heritage metadata exchange and promoted coordinated valorization of cultural history assets, but have had limited impact on diffusion and dissemination of each collection.

The state-of-the-art in automatic recognition of objects, actions, and other visual phenomena has advanced by leaps and bounds in the last five years (Russakovsky et al., 2015). This visual recognition technology can offer the potential of linking cultural tourists to the (currently inaccessible) collections of museums. Imagine the following scenario:

- A cultural tourist arrives at a destination rich in cultural heritage offerings.
- Our prototypical cultural tourist snaps a photo of an object or landmark of interest with his smartphone.
- After automatic recognition of the artwork or landmark, our tourist receives personalized, curated information about the object of interest and other cultural offerings in the area.

This type of scenario is realistic only if we have some way of easily recognizing a broad range of artworks. The challenges and barriers to this type of recognition technology have been studied in the past in the multimedia information analysis community (Cucchiara et al., 2012).

Recent breakthroughs in visual media recognition offer promise, but also present new challenges. One key challenging factor in the application of state-of-the-art classifiers is the data-hungry nature of modern visual recognition models. Even modestly sized Convolutional Neural Networks (CNNs) can have hundreds of millions of trainable parameters. As a consequence, they can require millions of *annotated* training examples to be effectively trained. The real problem then becomes the *cost* of annotation. Museum budgets are already stretched with *classical* curation requirements, adding to that the additional costs of collecting and annotating example media is not feasible.

Webly-supervised learning can offer solutions to the data annotation problem by exploiting abundantly available media on the web. This approach is appeal-

[1] https://europa.eu/cultural-heritage/

Del Chiaro, R., Bagdanov, A. and Del Bimbo, A.
NoisyArt: A Dataset for Webly-supervised Artwork Recognition.
DOI: 10.5220/0007392704670475
In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2019)*, pages 467-475
ISBN: 978-989-758-354-4

ing as it is potentially able to exploit the millions of images available on the web without requiring any additional human annotation. In our application scenario, for example, there are abundant images, videos, blog posts, and other multimedia assets freely available on the web. If the multimedia corresponding to specific instances of cultural heritage items can be retrieved and verified in some way, this multimedia can in turn be exploited as (noisy) training data. The problem then turns from one of a lack of data, to one of mitigating the effects of various types of noise in the training process that derive from its Webly nature (Temmermans et al., 2011; Sukhbaatar and Fergus, 2014).

In this paper we present a dataset for Webly-supervised learning specifically targeting cultural heritage artifacts and their recognition. Starting from an authoritative list of known artworks from DBpedia, we query Google Images and Flickr in order to identify likely image candidates. The dataset consists of more than 3000 artworks with an average of 30 images per class. A test set of 200 artworks with verified images is also included for validation. We call our dataset *NoisyArt* to emphasize its webly-supervised nature and the presence of label noise in the training and validation sets. *NoisyArt* is designed to support research on multiple types of webly-supervised recognition problems. Included in the database are document embeddings of short, verified text descriptions of each artwork in order to support development of models that mix language and visual features such as zero-shot learning (Xian et al., 2017) and automatic image captioning (Vinyals et al., 2017). We believe that *NoisyArt* represents the first benchmark dataset for webly-supervised learning for cultural heritage collections. [2]

In addition to the *NoisyArt* dataset, we report on baseline experiments designed to probe the effectiveness of pretrained CNN features for webly-supervised learning of artwork instances. We also describe a number of techniques designed to mitigate various sources of noise in training data, as well as techniques for identifying "clean" classes for which recognition is likely to be robust. These techniques provide several practical tools for building classifiers trained on automatically acquired imagery from the web.

The rest of the paper is organized as follows. In the next section we review recent work related to our contribution. In section 3 we describe the NoisyArt dataset designed specifically for research on Webly-supervised learning in museum contexts, and in section 4 we discuss several techniques to cope with noise in webly supervised data. In section 5 we present a range of experimental results establishing baselines for state-of-the-art methods on the NoisyArt dataset. We conclude with a discussion of our contribution in section 6.

[2]https://github.com/delchiaro/NoisyArt

2 RELATED WORK

In this section we review work from the literature related to the *NoisyArt* dataset and webly-supervised learning.

Visual Recognition for Cultural Heritage. Cultural heritage and recognition of artworks enjoys a long tradition in the computer vision and multimedia research communities. The Mobile Museum Guide was an early attempt to build a system to recognize instances from a collection of 17 artworks using photos from mobile phone (Temmermans et al., 2011). More recently, the Rijksmuseum Challenge dataset was published which contains more than 100,000 highly curated photos of artworks from the Rijksmuseum collection (Mensink and Van Gemert, 2014). The PeopleArt dataset, on the other hand, consists of high-quality, curated photos of paintings depicting people in various artistic styles (Westlake et al., 2016). The objectives of these datasets vary, from person detection invariant to artistic style, to artist/artwork recognition. A unifying characteristic of these datasets, is the high level of curation and meticulous annotation invested.

Another common application theme in multimedia analysis and computer vision applied to cultural heritage is personalized content delivery. The goal of the MNEMOSYNE project was to analyze visitor interest *in situ* and to then select content to deliver on the basis of similarity to recognized content of interest (Karaman et al., 2016). The authors of (Baraldi et al., 2015), on the other hand, concentrate on closed-collection artwork recognition and gesture recognition using a wearable sensor to enable novel interactions between visitor and museum content.

Webly-supervised Category Recognition. Early approaches to webly-supervised learning (long before it was called by that name), were the decontamination technique of (Barandela and Gasca, 2000), and the noise filtering approach of (Brodley and Friedl, 1999). Both of these approaches are based on explicit identification and removal of mislabeled training samples. A more recent approach is the noise *adaptation* approach of (Sukhbaatar and Fergus, 2014). This approach looks at two specific types of label noise – labelflip and outliers – and modifies a deep network architecture to absorb and adapt to them. A very re-

cent approach to webly-supervised training of CNNs is the representation adaptation approach of (Chen and Gupta, 2015). The authors, in this work, at first fit a CNN to "easy" images identified by Google, and then adapt this representation to "harder" images by identifying sub- and similar-category relationships in the noisy data.

The majority of work on webly-supervised learning has concentrated on category learning. However, the *NoisyArt* dataset we introduce in this paper is an instance-based, webly-supervised learning problem. As we will describe in section 3, instance-base learning presents different sources of label noise than category-based.

Landmark Recognition. The problem of landmark recognition is similar to our focus of artwork classification, since they are both *instance* recognition problems rather than *category* recognition problems. It is also one of the first problems to which webly-supervised learning was widely applied. The authors of (Raguram et al., 2011) use webly-supervised learning to acquire visual models of landmarks by identifying *iconic views* of each landmark in question. Another early work merged image and contextual text features to build recognition models for large-scale landmark collection (Li et al., 2009).

Artwork recognition differs from landmark recognition, however, in the diversity of viewpoints recoverable from web search alone. As we will show in section 3, the NoisyArt dataset suffers from several types of label bias and label noise which are particular to the artwork recognition context.

3 THE *NoisyArt* DATASET

NoisyArt is a collection of artwork images collected using articulated queries to metadata repositories and image search engines on the web. The goal of *NoisyArt* is to support research on webly-supervised artwork recognition for cultural heritage applications. Webly-supervision is an important feature, since in the cultural applications data can be acutely scarce. Thus, the ability to exploit abundantly available imagery to acquire visual recognition models would be a tremendous advantage.

We feel that *NoisyArt* can be well-suited for experimentation on a wide variety of recognition problems. The dataset is particularly well-suited to webly-supervised instance recognition as a weakly-supervised extension of fully-supervised learning. To support this, we provide a subset of classes with manually verified test images (i.e. with *no label noise*).

In the next section we describe the data sources used for collecting images and metadata. Then in section 3.2 we describe the data collection process and detail the statistics of the *NoisyArt* dataset.

3.1 Data Sources

To collect the *NoisyArt* dataset we exploited a range of publicly available data sources on the web.

Structured Knowledge Bases. As a starting point, we used public knowledge bases like DBpedia (Bizer et al., 2009; Mendes et al., 2011) and Europeana (Valtysson, 2012) to query, select, and filter the entities to be used a basis for *NoisyArt*. The result of this exercise was a reduced list of 3,120 artwork classes with Wikipedia entries and ancillary information for each one: title, descriptions, museum in which the artwork is preserved, artist information like name, birth and death date, description, and artistic movement.

DBpedia. DBpedia is the same source from which we retrieved metadata. For some artworks it also contains one or more images. We call this kind of images a *seed image* because it is unequivocally associated with the metadata of the artwork. Note, however, that though the association is reliable, some times the seed image is an image of the *artist* and not of the *artwork*.

Google Images. We queried Google Images using the title of each artwork and the artist name. For each query we downloaded the first 20 retrieved images. These images tend to be very clean, in particular for paintings, most of which do not have a background and tend to be very similar to scans or posters. For this reason the variability of examples can be poor: we can retrieve images that are almost identical, maybe with just different resolutions or with some differences in color calibration. Another issue with Google Image search results is the label flip phenomenon: searching for minor artworks by a famous artist can result in retrieving images of other artworks from the same artist. Outliers are also present in a small part for less famous artworks by less famous artists.

Flickr. Finally, we used the Flickr API to retrieve a small set of images more similar to real-world pictures taken by users. Due to its nature, the images retrieved from Flickr tend to be more noisy: the only supervision is by the end-users, and a lot of images (specially for famous and iconic artworks) do not contain the expected subject. For least famous artworks, the number of retrieved images is almost zero and can be full of outliers. For these reasons we only retrieve the first 12 images from each Flickr query in order to filter some of the outlier noise.

Discussion. In the end, Flickr images are the most

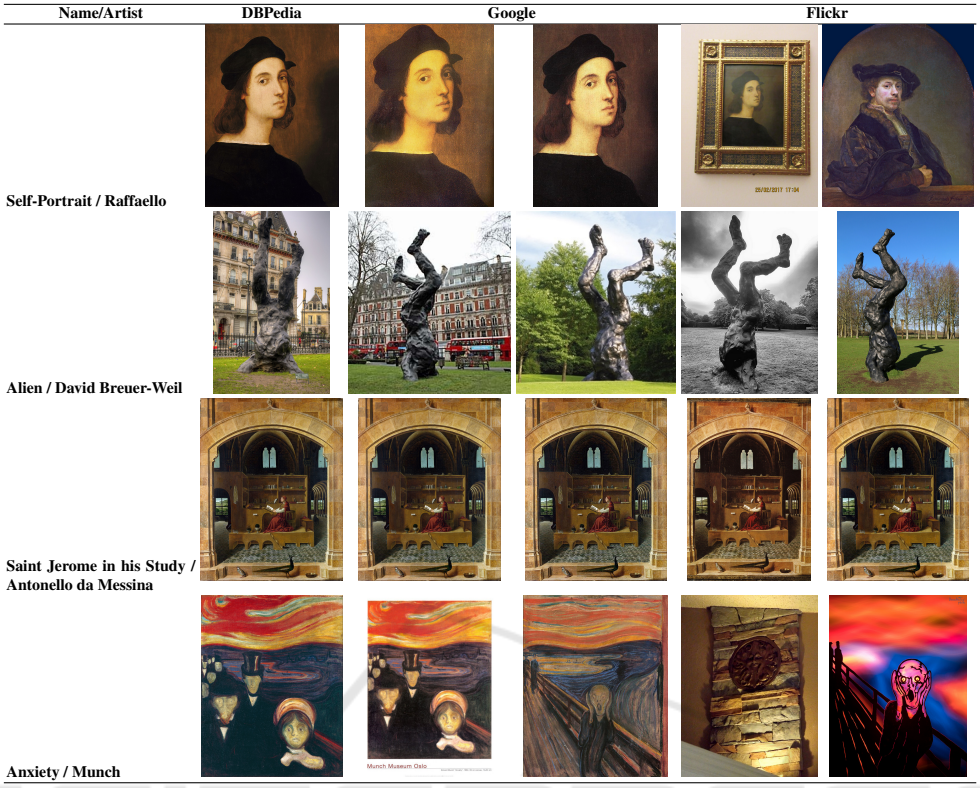

Figure 1: Sample classes and training images from the *NoisyArt* dataset. For each artwork/artist pair we show the seed image obtained from DBpedia, the first two Google Image search results, and the first two Flickr search results.

informative due to variety and similarity to real-world pictures. However a lot of them are incorrect (outliers). DBpedia seed images are the most reliable but are at most one per artwork. Google images are usually more consistent with the searched concept when compared to the Flicker ones, but normally present low variability.

3.2 Data Collection

From these sources we managed to collect 89,395 images for the 3120 classes, that became 89,095 after we pruned unreadable images and some error banners received from websites. Before filtering, each class contained a minimum of 20 images (from google) and a maximum of 33 (12 from Flickr and the DBpedia seed).

We could have used the seed images as a single-shot test set (pruning all the classes without the seed) but the importance of these images in the training phase joined to the inconsistency of seed in some classes led us to create a supervised test set using a small subset of the original classes: 200 classes containing more than 1,300 images taken from the web or from our personal photos. We have been careful not to use images from the training set. This test set is not balanced: for some classes we have few images, and some others have up to 12. Figure 2 illustrates some sample classes and images from our verified test set. Note the strong domain shift in these images with respect to those in the training set shown in figure 1.

Table 1: Characteristics of the *NoisyArt* dataset.

		(webly images)		(verified images)
	classes	**training**	**validation**	**test**
	2,920	65,759	17,368	0
	200	4,715	1,253	1,355
totals:	3,120	70,474	18,621	1,355

Finally, each artwork has a description and metadata retrieved from DBpedia, from which a single textual document was created for each class. These short descriptions were then embedded using doc2vec (Le and Mikolov, 2014) in order to provide a compact, vector space embedding for each artwork description. These embeddings are included to support research on zero-shot learning and other multi-modal approaches to learning over weakly supervised data.

In the end, *NoisyArt* is a multi-modal, weakly-supervised dataset of artworks with 3,120 classes and more than 90,000 images, 1,300 of which are human validated. Table 1 details the breakdown of the splits defined in *NoisyArt*.

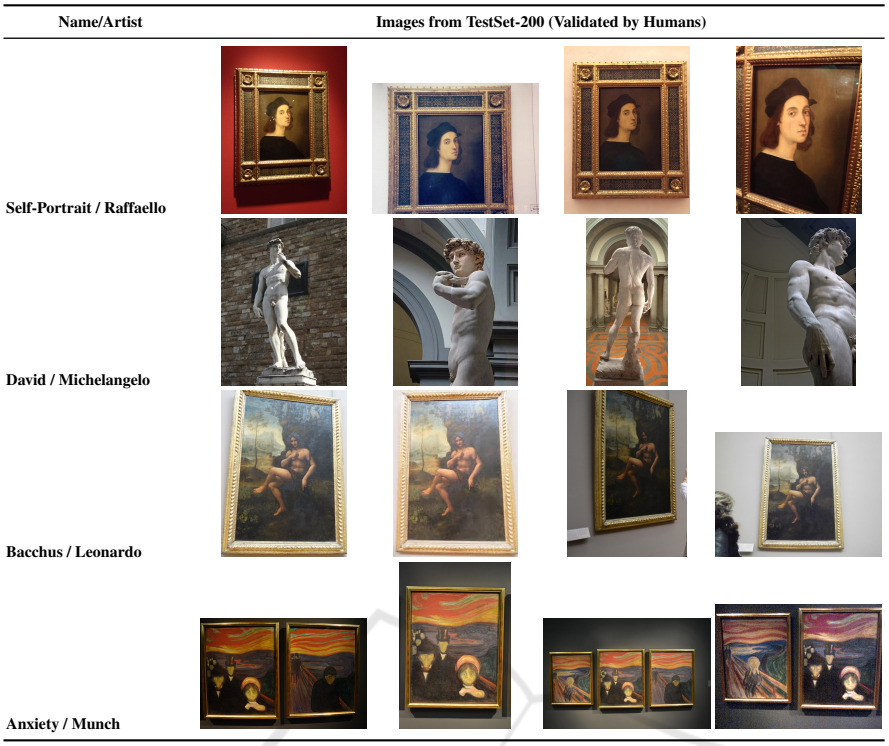

Figure 2: Sample verified test images from the *NoisyArt* test set. For a random sample of 200 classes we collected an additional set of images that we manually verified. Note the significant domain shift on these images with respect to those in figure 1.

3.3 Discussion

In figure 1 we give a variety of examples from the *NoisyArt* dataset. For each artwork we show: the seed image from DBpedia, the first two Google Image search results, and the first two results from Flickr. These examples show typical scenarios of this artwork instance recognition problem:

- **Best Case.** The second row of figure 1 contains pictures of a statue. For these kinds of objects it is usually much easier to retrieve images with a good level of diversity, both from Google and Flickr. This is due to the 360° access and thus the relative variety of viewpoints from which such artworks are photographed.
- **Lack of Diversity.** The first row of figure 1 is an example of an artwork for which Google retrieves images with extremely low variety, although in this case Flickr returns images with some diversity, but also outliers. In the third row we can observe an example for which both Google and Flickr failed to have diversity.
- **Labelflip.** In the fourth of figure 1 we see a pathology particular to our instance recognition problem: we are looking for images of a *not-so-famous artwork* (Anxiety) by a *famous* artist (Munch) who also made much more iconic artworks (like The Scream). In these cases the risk of labelflip is high, and in fact we retrieved from both Google and Flickr also images of *The Scream* (together with some correct and some outlier images).

These types of label noise in the *NoisyArt* dataset render it difficult to acquire robust visual models using webly supervision. In the next section we discuss techniques to mitigate or identify noise during training.

4 COPING WITH NOISY DATA

In this section we describe several techniques for mitigating and/or identifying label noise during training. First we describe the baseline classifier model used in all experiments.

4.1 Baseline Classifier Model

For all our experiments we use a shallow classifier based on image features extracted from CNNs pretrained on ImageNet. Figure 3 shows the architecture

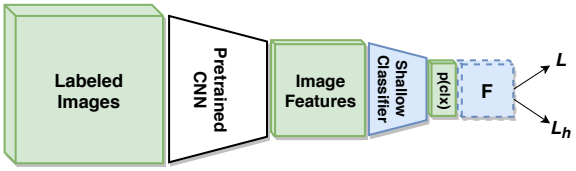

Figure 3: Classifier models used for webly-supervised experiments on *NoisyArt*. Green blocks represent data flowing through the network, blue ones components with trainable parameters. A CNN (pretrained on ImageNet) is used to extract features from training images, and then a shallow network with a single hidden layer and output layer are trained to predict class probabilities. The F matrix (see section 4.2) is used to model and absorb labelflip noise in the training set, and the loss function is either the cross entropy loss L or the weighted cross entropy loss L_h described in section 4.3.

of our networks. Given an input image $\mathbf{x}$, we extract a feature vector using the pretrained CNN and then we pass it trough a shallow classifier, consisting of a single hidden layer (with the same size as the extracted features) and an output layer that estimates class probabilities $p(c|\mathbf{x})$ for each of the 200 test classes.

The shallow classifier is then optionally followed by multiplication by a 200×200 *labelflip* matrix F (see section 4.2). For the baseline experiments F is set to the identity matrix.

Finally, the loss function used to train the shallow network weights is the cross entropy loss:

$$L(\mathbf{x}, y; \theta) = -\sum_c \mathbb{1}_y(c) p(c \mid \mathbf{x}),$$

where $\mathbb{1}_y(c)$ is the indicator function:

$$\mathbb{1}_y(c) = \begin{cases} 1 & \text{if } c = y \\ 0 & \text{otherwise.} \end{cases}$$

The alternate loss function L_h shown in figure 3 is described in section 4.3.

4.2 Labelflip Noise

Labelflip noise refers to images in the training set which are mislabeled as belonging to the *incorrect* class. This problem can be acute in instance recognition, for example when artists have works which are significantly more famous than their others and these famous works are often returned on queries. We experimented with the technique for labelflip absorption proposed in (Sukhbaatar and Fergus, 2014).

The main idea of labelflip absorption is to introduce a new fully connected layer without bias after the final softmax output (see the component F in figure 3). The weights of this layer, which we call F, are an $N \times N$ stochastic matrix, where N is the number of classes. Each row of F models the likelihood of confusing one class for any of the other classes. This matrix is initialized to the identity matrix and, at the start of training, the weights are locked (not trainable). After a number of training epochs (500 in our experiments), the weights are unlocked, allowing F to model class confusion probabilities and to spread out the probability mass from each class to common confusions for that class, thanks also to a trace regularization. At each training iteration the rows of F are reprojected onto the N-simplex to keep F stochastic. The result is that labelflip noise is absorbed into the F matrix, leaving the network free to learn on "clean" labels.

4.3 Entropy Scaling for Outlier Mitigation

The labelflip matrix described in the previous section attempts to compensate for class-level confusions during training. In this section we describe an alternate technique that performs soft outlier detection in order to weight training samples during training. Our hypothesis is that the class-normalized entropy of a training sample is an indicator of how confident the model is about a particular input sample.

The normalized entropy of a training sample $\mathbf{x}_i$ is defined as:

$$\hat{H}(\mathbf{x}_i) = -\frac{1}{C} \sum_c p(c \mid \mathbf{x}_i) \ln p(c \mid \mathbf{x}_i),$$

where C is a normalizing constant equal to the maximum entropy attainable for the given number of classes. When $\hat{H}(\mathbf{x}_i)$ is zero, the classifier is absolutely certain about $\mathbf{x}_i$; when it is one, the classifier has maximal uncertainty.

The entropy weighted loss is defined as:

$$L_h(\mathbf{x}, y; \theta) = -\sigma(\hat{H}(\mathbf{x})) \sum_c \mathbb{1}_y(c) p(c \mid \mathbf{x}),$$

where the normalized entropy is passed through a modified sigmoid σ function of the types illustrated in figure 4. This function is defined as:

$$\sigma(x; m, b) = \frac{1}{1 + e^{m(x-b)}}$$

so that the loss for training sample $\mathbf{x}$ is weighted inversely proportionally to the normalized entropy $\hat{H}(\mathbf{x})$.

4.4 Gradual Bootstrapping

The entropy scaling technique described in the previous section applies soft weights to the loss contributed by specific training samples. These weights are based

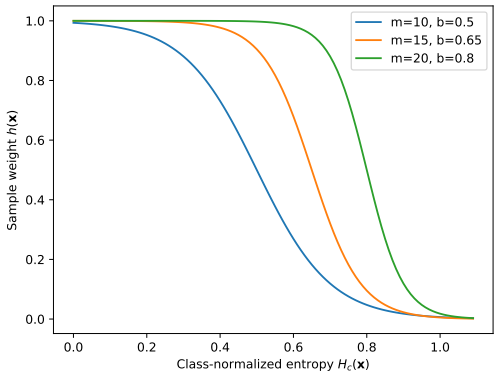

Figure 4: Modified sigmoid function used to calculate per-sample weights based on normalized entropy. The m parameter controls the steepness of the transition from 1.0 to 0.0, and the b parameter the point at which is begins its transition.

on an estimate of the class uncertainty. However, CNNs are known to produce highly-confident predictions even on outliers. Instead, here we propose a method for gradually bootstrapping during training by starting from highly reliable training examples, and sequentially introducing less reliable training data.

For *NoisyArt* we have the *seed images* acquired from DBpedia metadata records that can be used as an absolutely reliable image for each class. If there is no seed image, we use the first result returned by Google Image Search as the initial bootstrap image for each class. Training is performed for 80 epochs using only seed images, then the rest of the training images are added and training proceeds using entropy scaling as described in section 4.3. We expect that after acquiring a reliable model on seed images, entropy scaling will be more robust as the classifiers should be more conservative as they have been initially trained on a very reduced training set.

4.5 Identifying Problem Classes

The entropy measure used for noise mitigation in the approach described in section 4.3 can also be used to filter "problem classes" in the sense that the average class entropy on the test images is an indicator of classifier uncertainty. For each class we compute the average entropy, as measured by a trained shallow classifier model, for every training image in that class. We then rank the classes by decreasing average entropy. We then progressively remove these problem classes. This provides a practical technique for filtering unreliable classes from the final model.

Table 2: Networks used for image feature extraction in our experiments.

CNN	Reference	Feature	Size
ResNet-50	(He et al., 2016)	Global pool	2048
ResNet-101	(He et al., 2016)	Global pool	2048
ResNet-152	(He et al., 2016)	Global pool	2048
VGG16	(Simonyan and Zisserman, 2014)	FC7	4096
VGG19	(Simonyan and Zisserman, 2014)	FC7	4096

Table 3: Baseline and noise filtering results for webly-supervised recognition. We report accuracy (acc) and mean average precision (mAP) results on both the webly-supervised validation and fully-validate test sets.

	test		validation	
	acc	**mAP**	**acc**	**mAP**
ResNet-50 BL	67.01	54.14	76.46	64.10
ResNet-50 LF	67.90	55.83	76.54	63.54
ResNet-50 ES	**68.71**	57.42	76.46	63.74
ResNet-50 BS	68.27	**57.44**	75.98	62.83
ResNet-101 BL	67.31	54.82	76.54	63.33
ResNet-101 LF	67.08	55.58	**77.09**	**64.17**
ResNet-101 ES	67.16	56.60	76.38	63.56
ResNet-101 BS	68.27	57.41	76.78	63.46
ResNet-152 BL	67.60	54.92	76.70	63.66
ResNet-152 LF	66.72	54.66	76.46	63.02
ResNet-152 ES	67.16	56.06	76.70	64.16
ResNet-152 BS	67.38	55.81	76.22	62.90
VGG16 BL	64.72	50.61	74.86	60.82
VGG16 LF	64.65	50.62	73.74	59.23
VGG16 ES	64.80	51.17	75.42	61.65
VGG16 BS	66.27	52.52	74.38	60.07
VGG19 BL	62.80	48.98	73.50	59.42
VGG19 LF	61.33	46.53	73.07	57.84
VGG19 ES	61.92	48.43	72.87	58.34
VGG19 BS	63.99	51.14	72.63	58.21

5 EXPERIMENTAL RESULTS

In this section we report experimental results for a number of feature extraction and label noise compensation methods. All experiments were conducted using features extracted from CNNs pretrained on ImageNet, which are then fed as input to a shallow classifier (see figure 3). More specifically, we extracted features using the networks shown in table 2.

The shallow networks were trained with the Adam optimizer (Kingma and Ba, 2014) for 1500 epochs on the 200-class training set. We use a learning rate of 1e-4 and L2 weight decay with a coefficient of 1e-7. For experiments using entropy scaling, we use parameters $m = 20$ and $b = 0.8$ for the modified sigmoid function. After 1500 epochs, the model corresponding to the best classification accuracy on the webly-supervised validation set was evaluated on the verified test set.

5.1 Webly-supervised Classification

Table 3 gives results for all extracted features. For each extracted feature type we report results for:

- **Baseline (BL)**: the shallow network trained with no noise mitigation.
- **LabelFlip (LF)**: the shallow network trained with labelflip absorption as described in section 4.2.
- **Entropy Scaling (ES)**: the shallow network trained with entropy scaling as described in section 4.3.
- **BootStrapping (BS)**: the shallow network trained with gradual bootstrapping as described in section 4.4.

From table 3 we can draw a few conclusions. First of all, despite the high degree of noise in the training labels, even the baseline classifiers perform surprisingly well on the webly-supervised learning problem. All of the ResNet models achieve nearly 70% classification accuracy on the verified test set. The shallow classifier seems to be able to construct models robust to noise in the majority of classes.

Another interesting observation to be made from table 3 is that simpler models seem to perform better, on average. ResNet-50 and ResNet-101 outperform the more complex models. The VGG16 and VGG19 features perform significantly worse than all of the ResNet features.

All three of the noise mitigation techniques generally improve over the baseline shallow classifier, though not always by a significant margin. Our gradual bootstrapping technique described in section 4.4 generally yields the most consistent and significant improvement: about 2% improvement in accuracy and 3% in mAP over the baseline on ResNet-50 and ResNet-101 features.

Finally, results on the validation set are an unreliable fine-grained predictor of classifier performance on validated test data. Though the performance on the validation set between ResNet and VGG models is a reliable indicator, performance on the different ResNet models is too close to call.

5.2 Identifying Problem Classes

In figure 5 we show the improvement that can be gained by filtering classes with high average entropy. The figure plots classifier accuracy for all models with bootstrapping as a function of progressively filtered test sets (i.e. removing unreliable classes). Observe that the average class entropy is a reasonable measure of classifier reliability. After filtering only about 20% of the problem classes we can obtain an overall accuracy of better than 80% on the remaining ones for the ResNet models.

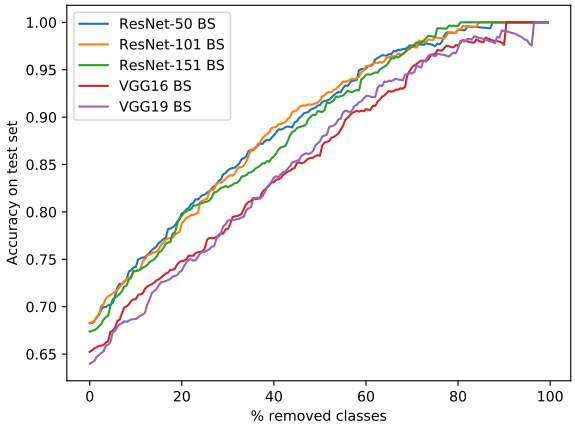

Figure 5: Filtering problem classes. We progressively remove classes with high entropy from the test set. Accuracy is plotted as a function of the number of remaining classes.

6 CONCLUSIONS AND FUTURE WORK

In this paper we described the *NoisyArt* dataset which is designed to support research on webly-supervised training for artwork recognition. *NoisyArt* contains more than 3,000 classes with over 90,000 corresponding images automatically collected from the web. Metadata and document embeddings are included for all artworks. A verified set of test samples for a subset of 200 classes is also provided.

Preliminary results on artwork recognition using shallow classifiers trained on features extracted with pretrained CNNs are encouraging. Baseline classifiers, with relatively simple networks and compact image features, achieve nearly 70% classification accuracy when trained on webly-supervised data. Noise mitigation techniques are able to improve performance, though the increase is at times marginal. Our technique for noise mitigation, a type of gradual bootstrapping, yields consistent improvements on most features.

We also described a technique for identifying problem classes (i.e. classes whose recognition is likely to be unreliable). Our approach is to use the average entropy of training samples, as measured by the outputs of the trained classifier, and filter those with high average entropy. Our results show that filtering only about 20% of classes can yield a dramatic increase in overall reliability.

Current and ongoing work includes research on

zero-shot learning using webly-supervised data, and a deeper investigation of entropy-based noise mitigation. We are also interested in investigating the potential for using entropy-based problem class identification as a means to articulate better queries for problem classes, leading to an iterative query-train-requery-retrain cycle in order to improve robustness.

REFERENCES

Baraldi, L., Paci, F., Serra, G., Benini, L., and Cucchiara, R. (2015). Gesture recognition using wearable vision sensors to enhance visitors' museum experiences. *IEEE Sens. J*, 15(5):2705–2714.

Barandela, R. and Gasca, E. (2000). Decontamination of training samples for supervised pattern recognition methods. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 621–630. Springer.

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., and Hellmann, S. (2009). Dbpedia-a crystallization point for the web of data. *Web Semantics: science, services and agents on the world wide web*, 7(3):154–165.

Brodley, C. E. and Friedl, M. A. (1999). Identifying mislabeled training data. *Journal of artificial intelligence research*, 11:131–167.

Chen, H. and Rahman, I. (2018). Cultural tourism: An analysis of engagement, cultural contact, memorable tourism experience and destination loyalty. *Tourism management perspectives*, 26:153–163.

Chen, X. and Gupta, A. (2015). Webly supervised learning of convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1431–1439.

Cucchiara, R., Grana, C., Borghesani, D., Agosti, M., and Bagdanov, A. D. (2012). Multimedia for cultural heritage: Key issues. In *Multimedia for Cultural Heritage*, pages 206–216. Springer.

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

Karaman, S., Bagdanov, A. D., Landucci, L., D'Amico, G., Ferracani, A., Pezzatini, D., and Del Bimbo, A. (2016). Personalized multimedia content delivery on an interactive table by passive observation of museum visitors. *Multimedia Tools and Applications*, 75(7):3787–3811.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *International Conference on Machine Learning*, pages 1188–1196.

Li, Y., Crandall, D. J., and Huttenlocher, D. P. (2009). Landmark classification in large-scale image collections. In *Computer vision, 2009 IEEE 12th international conference on*, pages 1957–1964. IEEE.

Mendes, P. N., Jakob, M., García-Silva, A., and Bizer, C. (2011). Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th international conference on semantic systems*, pages 1–8. ACM.

Mensink, T. and Van Gemert, J. (2014). The rijksmuseum challenge: Museum-centered visual recognition. In *Proceedings of International Conference on Multimedia Retrieval*, page 451. ACM.

Raguram, R., Wu, C., Frahm, J.-M., and Lazebnik, S. (2011). Modeling and recognition of landmark image collections using iconic scene graphs. *International journal of computer vision*, 95(3):213–239.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556.

Sukhbaatar, S. and Fergus, R. (2014). Learning from noisy labels with deep neural networks. *CoRR*, abs/1406.2080.

Temmermans, F., Jansen, B., Deklerck, R., Schelkens, P., and Cornelis, J. (2011). The mobile museum guide: artwork recognition with eigenpaintings and surf. In *Proceedings of the 12th International Workshop on Image Analysis for Multimedia Interactive Services*.

Valtysson, B. (2012). Europeana: The digital construction of europe's collective memory. *Information, Communication & Society*, 15(2):151–170.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2017). Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *IEEE transactions on pattern analysis and machine intelligence*, 39(4):652–663.

Westlake, N., Cai, H., and Hall, P. (2016). Detecting people in artwork with cnns. In *Computer Vision – ECCV 2016 Workshops*, pages 825–841.

Xian, Y., Schiele, B., and Akata, Z. (2017). Zero-shot learning - the good, the bad and the ugly. In *IEEE Computer Vision and Pattern Recognition (CVPR)*.