Profiling Online Auction Sellers Using Image-Editing Styles

Lin Yang
Google

Wei-Bang Chen, Chengcui Zhang, John K. Johnstone, Song Gao, and Gary Warner
University of Alabama at Birmingham

An automatic algorithm can extract image-editing styles to establish online auction seller profiles, which are fundamental to detecting account anomalies.

Product images serve an important role in online auction listings. As thriving businesses, online auction sites often host millions of concurrent auction listings. Where space is limited (such as on the page of auction search results), only product images are displayed to users as an overview of all auction listings. To stand out from competitors, veteran sellers often edit product images to attract potential buyers. Over time, many sellers have developed their own editing styles that recurrently appear in their image pool and are mostly distinct from other sellers, indicating a promising feature for seller profiling.

Seller profiling is fundamental for the detection of account anomalies, which are often related to fraudulent acts. Numerous online auction guides suggest that buyers watch for anomalies in a seller's auction listings (such as sudden changes in product categories, auction templates, and text fonts), because such anomalies often indicate account takeovers.[1] Researchers have proposed computational methods to encode such features and automate the detection of anomalies and frauds.[2] However, little previous work has leveraged product images, a major component of auction listings.

We developed an automatic algorithm that can extract image editing styles to establish seller profiles. First, we propose a preprocessing heuristic to promote repetitive visual elements in the editing areas of images and, thereby, bypass the separation of product and editing areas. Second, we rely on local feature matching to capture repetitive visual elements. The local features are designed to be invariant to various image transformations often seen in product images, including translation, scaling, partial occlusion, illumination, and color changes. We organize matched features into tracks, where each track consists of features of the same visual element. Tracking naturally separates edited and unedited images as well as images of distinct editing styles within a seller's image pool. Finally, we encode the editing styles for a seller using a bag-of-tracks model. We then weigh each track using two significance factors, measuring the frequency and spatial stableness of the corresponding visual element.

Using the encoded editing styles, we can profile sellers. Henceforth, an unseen product image can be compared to a seller profile to determine the likelihood of authorship (authorship identification). We can also compare seller profiles to unveil the same editing styles, which indicates the possibility of both accounts being operated by the same person or organization (multiple-account linking).

By comparing the proposed algorithm to previous work using a dataset collected from eBay, we tested the algorithm's performance. In all experiments, the proposed algorithm outperformed previous work by a large margin.

Characterizing Editing Profiles

By analyzing a large dataset of product images, we find that most editing falls into one of the following categories: frames and backgrounds, promotional or illustrational texts, and logos and watermarks. Figure 1 gives some examples.

Algorithmically, we define an *editing style* as repetitive visual elements within a seller's image pool. However, several factors complicate the extraction of editing styles. First, the algorithm must handle the separation between product and editing areas in images so that repetitive visual elements in product areas are excluded from editing styles. In most cases, product areas reside in the center of images, but exceptions exist (see Figure 1a). Even worse, some editing (especially logos and watermarks) also tends to appear in the center region

1070-986X/12/$31.00 © 2012 IEEE

Figure 1. Editing styles of online auction sellers. A seller might have (a) one consistent editing style, (b) one editing style with various minor modifications, (c) multiple distinct editing styles, or (d) a mixture of edited and unedited images. For the various transformations of editing styles—(e) translation, (f) scaling, (g) partial occlusion, and (h) color change—each subfigure shows two images from the same seller where the editing style undergoes a certain transformation. The within-seller variations and transformations make it challenging to extract editing styles.

of images, rendering location-based heuristics invalid. Second, sellers might not rigidly stick to one editing style for all product images. They might apply different editing styles to different images, while leaving others unedited (see Figures 1a through 1d). Therefore, a naive averaging over all product images is insufficient. Finally, editing styles are likely to undergo various transformations during the preparation of product images. For example, logos might be moved and scaled to avoid overlapping with products, and colors might be adjusted to enhance the overall image appearance (see Figures 1e through 1h). Therefore, global features are also inadequate.

The automatic algorithm we propose here addresses all these complicating factors. (See the "Related Work in Authorship Attribution" sidebar for previous work.)

Editing-Style Extraction

The input to our editing-style extraction algorithm is a seller's image pool. The algorithm output is a bag-of-tracks model encoding image editing styles for the seller.

Near-Duplicate Image Removal

Before we capture repetitive visual elements in a seller's image pool, we must separate the images' product and editing areas. It is not uncommon for different sellers to acquire the same raw product image from the Web but then apply their own unique editing to the image before embedding it in auction listings. In such cases, the product area is expected to be shared among sellers and must be excluded from an individual seller's editing styles even if it appears recurrently in his or her image pool.

Accurate segmentation of product areas for general product images is unlikely, given the variety of image content. Instead, we propose a preprocessing heuristic to promote repetitive visual elements in the editing areas of images and thereby bypass the separation of product and editing areas. We make the assumption that a seller tends to edit the same raw product images in the same style, resulting in the edited images being near duplicates. Therefore, we preprocess the image pool of each seller to remove near duplicates. By doing so, we effectively keep only images of different raw product views. Therefore, repetitive visual elements are unlikely to appear in product areas.

We remove near-duplicate images in two passes. In the first pass, each image is treated as a binary string and hashed by the MD5 Message-Digest Algorithm.[3] Images with the

Related Work in Authorship Attribution

Our work can be regarded as *authorship attribution*—"the science of inferring characteristics of the author from the characteristics of documents written by that author"[1]—in the image domain. Authorship attribution has a long history in the forensics field and a wide range of applications, such as authentication of disputed literary works and plagiarism detection. Researchers have developed various statistical methods to automatically extract author styles from a corpus of known works and have encoded the styles for authorship identification for unknown works. (Patrick Juola surveys these methods.[1])

In the image domain, however, the basic features for statistical methods (text tokens, vocabulary, and so on) are not readily available. The selection of visual features is an important step in the process, and it depends largely on applications.

Chengcui Zhang and her colleagues proposed an algorithm for clustering images attached to spam emails because images in the same cluster are likely to be created by the same spammer.[2] They took a multimodel approach and extracted various features from spam images, including color histograms, layout masks, texture features, and optical character recognition (OCR) texts. They applied two-level clustering to spam images, where the first level of images was based on visual features and the second level refined the clusters by textual features. The algorithm was evaluated on a large dataset of spam images, and it achieved a high accuracy. However, compared to the product images in online auctions, spam images mostly consist of text, which is intended to help them bypass text-based spam filters. Therefore, their algorithm is not readily adapted to the extraction of editing styles in product images.

Our work is most closely related to that of Liping Zhou, Wei-Bang Chen, and Chengcui Zhang,[3] who developed a framework for authorship identification for eBay images. For each seller, they encoded all images using edge maps and aligned them with a generalized Hough transform.[4] With image similarity determined by overlapping areas of edge maps, the images of each seller were clustered into distinct editing styles. The authors proposed three methods to encode the editing style for each cluster on the basis of edge and color features. They then evaluated the methods for authorship identification for 47 sellers and 3,980 images collected from eBay. In their experiments, edge features outperformed color features in accuracy, with a sacrifice in processing time. This articles compares our proposed algorithm to that edge-based method and demonstrates the superior performance of our approach in three experiments.

References

1. P. Juola, "Authorship Attribution," *Foundations and Trends in Information Retrieval,* vol. 1, no. 3, 2006, pp. 233–334.
2. C. Zhang et al., "A Multimodal Data Mining Framework for Revealing Common Sources of Spam Images," *J. Multimedia,* vol. 4, no. 5, 2009, pp. 313–320.
3. L. Zhou, W.-B. Chen, and C. Zhang, "Authorship Detection and Encoding for eBay Images," *Int'l J. Multimedia Data Eng. and Management,* vol. 2, no. 1, 2011, pp. 22–37.
4. L.A. Fernandes and M.M. Oliveira, "Real-Time Line Detection through an Improved Hough Transform Voting Scheme," *Pattern Recognition,* vol. 41, no. 1, 2008, pp. 299–314.

same MD5 checksum are bit-wise identical, so we remove all but one from the image pool. The first pass efficiently detects and removes identical images, which appear frequently in online auction listings. In the second pass, the remaining images are converted to HSV (hue, saturation, value) color space and downscaled to 100×100. Using pixel-wise comparison, we determine whether two images are near identical if they share a majority of similar-value pixels.

In this study, we define similar-value pixels as ones that differ by less than 15 percent in all HSV channels, and we define two images as near identical if they share more than 80 percent of similar-value pixels. Most near-identical images by this standard correspond to the same image undergoing slightly different transformations (image compression, scaling, and so forth). Images are scanned sequentially and removed if near identical to any previous images.

Feature Matching

We rely on local feature matching to capture repetitive visual elements in a seller's image pool. Local features have been widely used for image-matching tasks because of their superior resistance to various image transformations.

We adopt the principal component analysis (PCA) scale invariant feature transform (SIFT) algorithm for local feature detection and encoding.[4] PCA-SIFT uses a SIFT detector[5] to detect feature points as local extrema in a series of difference-of-Gaussian (DoG) functions in the scale space. For each feature point, PCA-SIFT computes a gradient map over the local image patch and projects the vectorized gradient map to a 36-dimensional eigenspace as the feature descriptor for this feature point. For a regular-sized 300 × 300 pixel product image, PCA-SIFT usually detects and encodes several hundred feature points.

Encoded by PCA-SIFT feature descriptors, the match for a feature point is identified as its nearest neighbor in the feature space measured by Euclidean distance. We adopt a k-dimensional tree-based approximation algorithm—Best Bin First (BBF)—for a sublinear nearest-neighbor search.[6] The Euclidean distance between two nearest neighbors is subject to a threshold $\sigma = 3{,}000$,[4] below which the two feature points are determined to be a match.

Finally, a geometric constraint verifies the correctness of features matches. We adopt a robust estimator, the Random Sample Consensus (Ransac) algorithm,[7] to iteratively recover the epipolar geometry[8] between two images and find the largest subset of feature matches that are geometrically consistent. After geometric verification, few false matches remain.

We apply feature matching to all pairs of images within a seller's image pool. In our experimental dataset, each seller's image pool contains up to 200 images. Feature detection and encoding for 200 images and matching for 19,900 image pairs take about the same amount of time, totaling less than one hour per seller.

Feature Tracking

From feature matches among all pairs of images, we can capture repetitive visual elements in a seller's image pool. We organize the matched features into tracks, where a track is a connected set of matched features across multiple images. Tracks containing more than one feature point from the same image are deemed to be inconsistent and removed. Thus, each track corresponds to a repetitive visual element in the image pool.

We use tracks to encode editing styles. However, not all tracks carry equal weight, because some visual elements are less reliable than others in characterizing a seller's editing style. Thus, we assign each track two weights measuring the significance of the corresponding visual element.

The first weight, w_f, measures the frequency of the corresponding visual element. For sellers with multiple (and probably unbalanced) editing styles, a frequently used editing style should certainly be weighted more heavily than a rarely used one. We can easily derive the frequency of a visual element from the length of the track. Let a seller's image pool be denoted by $I = \{I\}$, and let a track, represented by a set of matched features, be denoted by $t = \{f\}$. Then, we define w_f as follows:

$$w_f = \frac{|t|}{|I|} \tag{1}$$

By definition, $w_f \in (0, 1]$,with a higher value corresponding to a more frequently used editing style.

The second weight, w_e, measures the spatial stableness of the corresponding visual element. In practice, we find that an editing style is more reliable if it has a fixed relative location across images. The measure of spatial stableness is especially useful when some tracks are falsely picked up in product areas. We introduced a preprocessing heuristic to demote tracks in product areas earlier. However, the heuristic cannot handle cases where the product area's relative location changes across images. In such cases, the images are not near duplicates by pixel-wise comparison, and tracks are inevitably picked up in product areas. However, the change in relative locations enables w_e to demote the significance of such tracks.

We define a track's weight w_e in terms of the entropy of the spatial distribution of all feature points in the track. For each feature point, we compute its relative location in the original image and project the relative location to a unit square. We then subdivide the unit square to 4 × 4 blocks. By counting the number of feature points in each block, we can form a 16-dimensional histogram measuring the spatial distribution of all feature points in the track. The histogram is normalized to a unit L1-norm to form a probability distribution. Let it be denoted by $H = \{p_1, \ldots, p_{16}\}$. Then, we define w_e as follows:

$$\text{entropy}\ (H) = -\sum_{i=1}^{16} p_i \log(p_i) \tag{2}$$

$$w_e = 1 - \frac{1}{\lambda}\text{entropy}\ (H) \tag{3}$$

where $\lambda = 4$ is a normalization factor to ensure that $w_e \in [0, 1]$. At the two extremes, $w_e = 0$ indicates that the track's feature points are completely randomly located, whereas $w_e = 1$ indicates that they are consistently located at one relative location.

Bag-of-Tracks Model

We encode a seller's editing styles using a bag-of-tracks model—that is, an unordered collection of tracks, each associated with two significance factors. We match PCA-SIFT features only within each seller's image pool. Therefore, each seller has a different dictionary of tracks. To match tracks across sellers, we assign each track a descriptor that is the centroid of descriptors for all feature points in the track. Hence, a seller can be represented by $u = \{p_1, \ldots, p_m\}$, where m is the number of tracks in the image pool of seller u, and p_i encodes the ith track by a 36-dimensional descriptor. The two functions w_f and w_e are associated with each track p_i. Each of the two functions takes a track as input and returns a scalar value in the [0, 1] range, measuring its frequency and spatial stableness, respectively. The bag-of-tracks model encodes all editing styles for a seller in one model. This is in contrast to previous work by Liping Zhou, Wei-Bang Chen, and Chengcui Zhang,[9] which required a seller's editing styles to be grouped and modeled separately.

An image has the same form of encoding by local features: $I = \{q_1, \ldots, q_n\}$, where n is the number of feature points in the image, and q_j encodes the jth feature point by a 36-dimensional descriptor. Therefore, sellers and images can be treated as homogeneous entities, and the same matching procedure we described earlier can be applied to match two sellers or an image and a seller.

In general, let two entities be denoted by $u = \{p_1, \ldots, p_m\}$ and $v = \{q_1, \ldots, q_n\}$. According to the matching procedure we described earlier, the set of matches between the two entities are defined by

$$M(u, v) = \{(p, q) \mid p \in u, q \in v, \text{ s.t. } p = NN_u(p,), \|p - q\| \leq \sigma\} \quad (4)$$

where function NN_u takes a query point and returns its nearest neighbor in the collection of u. Obviously, a large set of matches indicates that entities u and v share many visual elements. When one or both of u and v refer to sellers, the shared visual elements are mostly in editing areas. Therefore, we can use M to measure their overlap of editing styles.

For image-to-seller matching, let $u = \{p_1, \ldots, p_m\}$ be a seller and $v = \{q_1, \ldots, q_n\}$ be an image. Their overlap of editing styles is measured by

$$\text{overlap}(u, v) = \sum_{(p,q) \in M} w_f(p) w_e(p) \quad (5)$$

For seller-to-seller matching, let both $u = \{p_1, \ldots, p_m\}$ and $v = \{q_1, \ldots, q_n\}$ be sellers. Their overlap of editing styles is measured by

$$\text{overlap}(u, v) = \sum_{(p,q) \in M} w_f(p) w_e(p) w_f(q) w_e(q) \quad (6)$$

In both cases, the significance factors for tracks are applied whenever applicable to better quantify the contribution of matched tracks. Equations 5 and 6 serve as the basic mechanism for authorship identification and multiple-account linking, respectively.

Experiments

To evaluate our algorithm, we collected an experimental dataset from eBay. We started with the "Computer Accessories" category on eBay and crawled the first 50,000 concurrent auction listings (the maximum number allowed by the eBay search engine). For each distinct seller in the auction listings, we downloaded basic seller information: username, feedback score, positive feedback rate, and whether the seller was a "top-rated seller" (a metric defined by eBay for sellers with a track record of great service). Finally, we crawled the transaction history for each seller and downloaded product images from his or her most recent auction listings, up to three months or 200 images, whichever came first. In total, we downloaded 53,300 images from 637 sellers.

The dataset was subject to near-duplicate image retrieval, after which 23,490 images remained. For the purpose of editing-style extraction, we need to maintain a minimum count of images for each seller to capture repetitive visual elements. Thus, we discarded sellers who had less than five images left after near-duplicate image retrieval. The remaining, processed dataset consisted of 398 sellers and 17,914 images, which is several times larger than that used by Zhou, Chen, and Zhang.[9]

For each seller, we extracted images with editing styles and grouped them by distinct editing styles. Each group needed to contain at least two images because a single image with unique editing is not considered a "style." The same editing style can be shared

Table 1. Average V-measures achieved by our algorithm and the Zhou algorithm[9] under different parameter settings.

Experimental settings	V-measures				
	$\beta = 0.25$	$\beta = 0.50$	$\beta = 1.00$	$\beta = 2.00$	$\beta = 4.00$
$\theta = 5.0$	0.7498	0.7493	0.7501	0.7524	0.7554
$\theta = 6.0$	0.7531	0.7525	0.7531	0.7548	0.7571
$\theta = 7.0$	0.7569	0.7566	0.7569	0.7582	0.7599
$\theta = 8.0$	0.7523	0.7523	0.7531	0.7547	0.7568
$\theta = 9.0$	0.7538	0.7542	0.7553	0.7570	0.7592
$\theta = 10.0$	0.7527	0.7527	0.7534	0.7549	0.7569
Zhou algorithm	0.2011	0.2052	0.2120	0.2215	0.2325

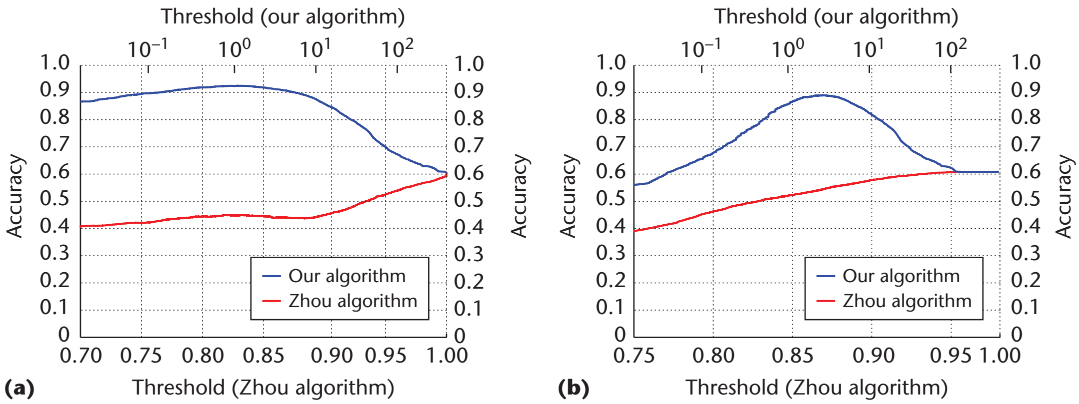

Figure 2. Accuracy for (a) within-seller and (b) cross-seller authorship identification. The proposed algorithm achieves high accuracy, whereas the Zhou algorithm[9] did very poorly for both within-seller and cross-seller identification.

among multiple sellers. Therefore, we also inspected editing styles across sellers and merged the same editing styles from different sellers. The manual grouping and merging of editing styles serves as the ground truth for our experiments.

We preformed three experiments: editing-style grouping, authorship identification, and multiple-account linking. In all three, we compared the proposed algorithm to the edge-based method from Zhou, Chen, and Zhang.[9] (See the "Related Work in Authorship Attribution" sidebar for more details.) We followed the parameter setting suggested in their original paper because they validated their algorithm on the same type of data (eBay product images). In all three experiments, the proposed algorithm outperformed the Zhou algorithm by a large margin.

In the editing-style-grouping and authorship-identification experiments, we randomly divided a set of 398 sellers into two equal subsets. We used one subset to train the algorithm, generated performance analysis (see Table 1 and Figure 2), and found the optimal parameter setting. We then applied the algorithm with the optimal parameter setting to the other subset to verify its robustness.

Editing Style Grouping

In the first experiment, we evaluated our algorithm's robustness by separating edited and unedited images and grouping images by distinct editing styles within a seller's image pool.

Based on the bag-of-tracks model, we established a metric between two images in a seller's image pool that measures their overlap in editing styles. After we encoded a seller using this approach, we were able to represent each image in the seller's image pool with a subset of the tracks that contain feature points in the image. Let seller u be represented by a bag of tracks $u = \{p_1, \ldots, p_m\}$, and let two of the seller's images be represented by their corresponding subsets of tracks: $I \subseteq u$, $J \subseteq u$. We use the intersection of the two subsets of tracks to measure their overlap in editing styles, weighted by their significance factors:

$$\text{overlap}(I, J) = \sum_{p \in I \cap J} w_f(p) w_e(p) \quad (7)$$

Note that the image-to-image matching assumes a form similar to the image-to-seller and seller-to-seller matching we described earlier.

We impose a threshold θ on the overlap scores, above which two images are considered to share the same editing style. Images are thereby grouped by connected components. Each connected component corresponds to a distinct editing style. We consider ungrouped images to be unedited, and they form another group. By varying the threshold θ, we can obtain different clustering results within each seller's image pool.

Using the ground truth we collected for editing-style grouping, we evaluated the clustering results generated by the algorithm using the V-measure,[10] which is an entropy-based method that measures the quality of clustering results. Given two clustering results over the same dataset, one ground truth and the other to be evaluated, the V-measure evaluates the latter by the criteria of homogeneity (h) and completeness (c). It combines the two criteria by their harmonic mean:

$$V_\beta = \frac{(1+\beta) \times h \times c}{\beta \times h + c} \tag{8}$$

where β controls the relative weight of completeness over homogeneity.

We vary the values for θ and β in reasonable ranges in their own domains. For each parameter setting, we computed and averaged V-measures from the seller training set. The average V-measures achieved by our algorithm are consistent, ranging between 0.74 and 0.76, as Table 1 shows. We then applied the optimal threshold $\theta = 7.0$ to the seller validation set. The resulting V-measures are {0.7348, 0.7329, 0.7315, 0.7308, 0.7309} for β in {0.25, 0.5, 1.0, 2.0, 4.0}.

By comparison, the V-measures achieved by the Zhou algorithm[9] are poor, ranging between 0.20 and 0.24. In their results, we find many false clusters that include images with no editing style. These images are grouped because the products are similar (such as images of different laptop batteries). These similarities lead to similar edge maps, upon which the images are falsely grouped.

Editing-style grouping had a profound impact on the subsequent steps in the Zhou algorithm because a separate model is built for each group of images to summarize the editing style. The models for editing styles, based on which authorship identification and multiple-account linking are conducted, are therefore contaminated. We believe that the Zhou algorithm's unsatisfactory performance in this and the subsequent experiments is rooted in editing-style grouping.

Authorship Identification

In this experiment, we identified the authorship for an unseen image by matching its editing style to the database of sellers encoded by their editing styles. To generate training data (database) and testing data (unseen images), we performed a five-fold cross validation on the experimental dataset. For each seller, we randomly partitioned the seller's image pool into five roughly equal subsets. During the ith fold of cross validation, we retained the ith subset from each seller as testing data and used the other four subsets from each seller as training data. The training data was processed by our algorithm to encode each seller with a bag-of-tracks model. Our algorithm matched the editing style of each image in the testing data to the encoded seller models. The image authorship was identified as the best matching seller (or unknown, if no match was found).

For this experiment, ground-truth collection for authorship identification seemed trivial because we knew the ownership for all the images. However, authorship cannot always be identified solely on the basis of image editing styles, because many images do not have editing styles and some editing styles are shared among multiple sellers. In this study, we are interested in authorship identification based on editing styles. Therefore, the ground truth is adapted accordingly. If an image does not have an editing style, its ground-truth authorship is "unknown." If an image has an editing style, its ground-truth authorship is its owner. However, if the editing style is shared by other sellers, then its ground-truth authorship is extended to the set of all involved sellers. Assigning the image to any seller in the set is deemed correct.

We conducted this experiment in two settings: within-seller matching and cross-seller matching.

For within-seller matching, a test image is only matched to its owner's model. The matching score is subject to a threshold θ_1, above which the authorship is identified as the owner (otherwise unknown). In practice, we can apply the mechanism to monitor new auction listings for anomalies. Images embedded in new auction listings can be constantly matched to the owner's established editing styles. An "unknown" image might indicate a breach in editing styles, which triggers an alarm for account takeover.

For cross-seller matching, a test image is matched to all seller models. The discriminative power of the bag-of-tracks model is evaluated in a more challenging setting. The best matching score is subject to a threshold θ_2, above which the authorship is identified as the best matching seller (otherwise unknown).

On the seller training set, our performance evaluation is based on accuracy, which we define as the percentage of correctly labeled images among all test images. We set he thresholds θ_1 and θ_2 experimentally; that is, we varied each threshold from the minimum to the maximum of all the matching scores generated by the test images, producing a complete accuracy curve, from which we obtained the optimal setting.

Figure 2 shows the accuracy curves for within-seller and cross-seller authorship identification for both our proposed algorithm and the Zhou algorithm. For within-seller authorship identification, the proposed algorithm achieves the highest accuracy at 92.36 percent, when $\theta_1 \approx 1.0$. Applying this threshold to the validation set of sellers yields an accuracy of 91.67 percent. For cross-seller authorship identification, the proposed algorithm achieves the highest accuracy at 88.86 percent, when $\theta_2 \approx 4.0$. Applying this threshold to the validation set of sellers yields an accuracy of 88.22 percent.

In both within-seller and cross-seller identification, few false positives occurred (images with no editing style that are falsely assigned an owner) thanks to the high precision offered by local feature matching. Under the optimal parameter setting, the false-positive rates for within-seller and cross-seller identification are 4.34 and 5.05 percent, respectively.

The Zhou algorithm did very poorly for both within-seller and cross-seller identification. Actually, their accuracy curves peak when all the test images are blindly assigned the label "unknown"—that is, the thresholds are set equal to the maximum matching scores.

Multiple-Account Linking

The goal of multiple-account linking is to unveil the same ownership among multiple seller accounts. Ground-truth collection is difficult because online auction sites do not reveal true seller identities. Actually, as Bezalel Gavish and Christopher Tucci point out,[11] complete information on true seller identities might even be unavailable to the online auction sites themselves because fraudsters often open multiple seller accounts disguised under different credit card information.

In this study, we are interested in the linking of multiple accounts revealed by image editing styles. Therefore, we define ground truth differently: two seller accounts are linked if and only if a subset of their images share the same editing style. Nevertheless, such accounts are indeed likely to share the same ownership, and we validated this hypothesis in the second part of the experiment. We collected ground-truth data by manually grouping distinct editing styles within each seller's image pool, and then merging the same editing styles across sellers. Out of all 79,003 seller pairs (among 398 sellers), 71 pairs were linked according to the ground-truth criterion. Due to the sparsity of links, we do not separate training and validation sets for this experiment.

We evaluated the performance of multiple-account linking in terms of precision and recall. Loosely speaking, precision measures how many links generated by the algorithm are correct according to the ground truth, and recall measures how many links in the ground truth are extracted by the algorithm. Mathematical definitions are as follows:

$$\text{precision} = \frac{\#\text{ correct links by algorithm}}{\#\text{ links by algorithm}} \tag{9}$$

$$\text{recall} = \frac{\#\text{ correct links by algorithm}}{\#\text{ links by ground truth}} \tag{10}$$

As we discussed earlier, the output of seller-to-seller matching is a score measuring their overlap of editing styles. By imposing a threshold θ on overlap scores, we can assign each pair of sellers (u, v) a concrete link or no-link label and thereby calculate the precision and recall rate:

$$\text{link }(u,v) = \begin{cases} \text{true}, & \text{overlap }(u,v) \geq \theta \\ \text{false}, & \text{otherwise} \end{cases} \tag{11}$$

Depending on practical requirements, we can achieve different precision-recall values by varying θ. Figure 3 shows the complete precision-recall curve formed by varying the threshold from the maximum to the minimum of all overlap scores (blue curve). Notice that the precision remains perfect (100 percent) within a large range of recalls (up to 63 percent), and the precision remains above 90 percent even after an 85 percent recall rate is achieved. The high precision is especially impressive considering that the target set (71 pairs of linked sellers by ground truth) is extremely small compared to the entire set of 79,003 seller pairs. Yet, the proposed algorithm successfully captures a majority of the targets without introducing many false positives.

In this experiment, we also demonstrated the improvement facilitated by track weighting. Each track is weighted by two significance factors, measuring the frequency and spatial stableness of the corresponding visual element. By default, the significance factors are used to adjust the contribution of each matched track in determining the overlap in editing styles between two sellers (see Equation 6). Without track weighting, Equation 6 is reduced to

$$\text{overlap}\ (u, v) = |M| \tag{12}$$

where the overlap in editing styles between two sellers is simply measured by the count of their matched tracks. We follow the same procedure, but we replace Equation 6 with Equation 12 to generate a second precision-recall curve for multiple-account linking (green curve in Figure 3). The precision drops sharply for the same recall rates. In certain recall ranges, the drop is more than 10 percent. These results, therefore, justify the proposed weighting scheme for tracks.

Once again, the Zhou algorithm does very poorly (see the red curve in Figure 3). Through the entire recall range, the precision remains at approximately 0.1 percent, which is barely better than that generated by random selection (71/79,003).

Finally, we validated the earlier hypothesis that accounts sharing the same editing styles are likely to be operated by the same person or organization. With true seller identities missing, we manually inspected linked sellers and collected ownership evidence from other sources. We found two sources to be useful in general. One is the basic seller information crawled from online auction sites, from which we spot unusual resemblances. For example, if two sellers share an uncommon substring in their

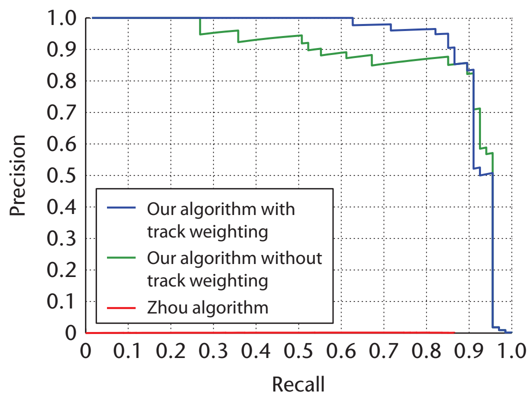

Figure 3. Precision and recall rate for multiple-account linking. The blue curve shows the proposed algorithm's precision-recall curve. Without track weighting, the proposed algorithm's precision drops sharply for the same recall rates (green curve). The Zhou algorithm's[9] precision remains at about 0.1 percent for the entire range of recall.

usernames and/or they use the same profile picture, then strong evidence exists that they share the same ownership. The other source of evidence is image watermarks, especially those that spell corporate names or website URLs. We consider two sellers to have the same ownership if they embed the same watermark in a majority of their images. (The requirement for quantity is to eliminate false-positive cases in which sellers occasionally copy images from each other or from the Web.) Under these criteria, we were able to find evidence for 34 out of the 71 pairs of linked sellers (47.89 percent). Because we only pursue the strongest evidence to achieve high credibility, we considered 47.89 percent to be the lower bound of the true-positive rate.

For the second part of the experiment, we tuned the threshold for linking two accounts to $\theta = 2.5$ to achieve a high precision of 90 percent. We linked sellers using Equation 11 and grouped linked sellers into clusters by connected components. In total, we were able to link 25 sellers and form 14 connected components (clusters) among them. Figure 4 shows the clusters and visualizes each cluster by sampling images from corresponding sellers.

For each cluster of sellers, we collected evidence from the two sources we mentioned earlier and determined if they shared the same ownership. Table 2 shows the validation results with evidence. Each row shows the validation result for one cluster of linked sellers and

Figure 4. Multiple-account linking. Each subfigure, (a) through (n), shows one cluster of seller accounts sharing common editing styles. Each account is represented by one sample image. Accounts in the same cluster are likely to be operated by the same person or organization.

Table 2. Same-ownership validation for linked seller accounts.

Cluster	Validated	Evidence
a	Y	Same watermark spelling hotsellnow.com
b	Y	Located in the same city; same profile picture
c	N	
d	Y	enessy_llc versus enessysales
e	N	
f	N	
g	Y	ibestbuy262 versus hey262mobile
h	Y	6ubuy6_8 versus 6ubuy6
i	N	
j	N	
k	N	
l	Y	sureelectronics versus sureelectronics1
m	Y	my_gadgetgarage versus mygadgetgarage
n	Y	apple-accessories-usa versus mac-accessories-usa

evidence for the same ownership, if any. We labeled the clusters in the same manner as in Figure 4. If the evidence is a similarity in usernames, it is simply expressed as ''username1 versus username2.''

For eight out of the 14 clusters, we were able to find evidence for the same ownership. Of course, our sources of evidence are by no means complete. During the evidence search, we noticed weaker signals for several unproven clusters, such as similarly structured product inventories or multiple identical product images. Nevertheless, the ratio of validation is impressive, demonstrating this image editing style to be a valuable clue for multiple-account linking.

Future Work

This article introduces image-editing style as a new feature for profiling online auction users. Initial experiments on a real-world dataset showed promising results. However, the power of this method is limited by its specific data requirements. For example, many online auction users have not established editing styles in their product images. Future work might consider a combination of multiple features such as a user's text, images, and transaction history to generate a more robust profile. **MM**

Acknowledgments

The work of Lin Yang and John Johnstone was supported in part by a Google Research Award.

References

1. S. Wright, *Don't Get Burned on eBay: How to Avoid Scams and Escape Bad Deals,* O'Reilly Media, 2006.
2. D.H. Chau, S. Pandit, and C. Faloutsos, "Detecting Fraudulent Personalities in Networks of Online Auctioneers," *Proc. European Conf. Principles and Practice of Knowledge Discovery,* LNCS 4213, Springer, 2006, pp. 103–114.
3. W. Stallings, *Cryptography and Network Security,* Prentice Hall, 2005.
4. Y. Ke and R. Sukthankar, "PCA-SIFT: A More Distinctive Representation for Local Image Descriptors. *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (CVPR 04), IEEE CS Press, 2004, pp. 506–513.
5. D.G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *Int'l J. Computer Vision,* vol. 60, no. 2, 2004, pp. 91–110.
6. J. Beis and D. Lowe, "Shape Indexing Using Approximate Nearest-Neighbour Search in High-Dimensional Spaces," *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (CVPR 97), IEEE CS Press, 1997, pp. 1000–1006.
7. M.A. Fischler and R.C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Comm. ACM,* vol. 24, no. 6, 1981, pp. 381–395.
8. R.I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision,* Cambridge Univ. Press, 2000.
9. L. Zhou, W.-B. Chen, and C. Zhang, "Authorship Detection and Encoding for eBay Images," *Int'l J. Multimedia Data Eng. and Management,* vol. 2, no. 1, 2011, pp. 22–37.
10. A. Rosenberg and J. Hirschberg, "V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure," *Proc. Joint Conf. Empirical Methods in Natural Language Processing and Computational Natural Language Learning,* Assoc. Computational Linguistics, 2007, pp. 410–420.
11. B. Gavish and C.L. Tucci, "Reducing Internet Auction Fraud," *Comm. ACM,* vol. 51, no. 5, 2008, pp. 89–97.

Lin Yang is a software engineer at Google. His research interests include mining crowd wisdom from Web photo collections and mining crowd wisdom from the Web. Yang has a PhD in computer science from the University of Alabama at Birmingham. Contact him at galabing@gmail.com.

Wei-Bang Chen is a PhD candidate in the Knowledge Discovery and Data Mining Lab of the Department of Computer and Information Sciences at the University of Alabama at Birmingham. His research interests include bioinformatics, multimedia data mining, and network security. Chen has an MS in genetics from National Yang-Ming University, Taiwan, and an MS in computer science from the University of Alabama at Birmingham. Contact him at wbc0522@cis.uab.edu.

Chengcui Zhang is an associate professor of computer and information sciences at the University of Alabama at Birmingham. Her research focuses on multimedia data mining. Zhang has a PhD in computer science from Florida International University. Contact her at zhang@cis.uab.edu.

John K. Johnstone is an associate professor in the Department of Computer and Information Sciences at the University of Alabama at Birmingham. His research interests include shape modeling and computer vision. Johnstone has a PhD in computer science from Cornell University. Contact him at jj@cis.uab.edu.

Song Gao is a PhD student in the Knowledge Discovery and Data Mining Lab of the Department of Computer and Information Sciences at the University of Alabama at Birmingham. His research interests include data mining, machine learning, and multimedia database systems. Gao has an MS in computer science from Chongqing University of Posts and Telecommunications, China. Contact him at gaos@cis.uab.edu.

Gary Warner is the director of research in computer forensics at the University of Alabama at Birmingham. He also serves on the Technology Committee of the US Federal Bureau of Investigation's Digital PhishNet and cochairs the Working with Law Enforcement Committee of the Anti-Phishing Working Group. His research interests include spam and phishing, malware, botnets, and cyber-warfare. Warner has a BS in computer information sciences from the University of Alabama at Birmingham. Contact him at gar@cis.uab.edu.