EXPLORING “ARTIST IMAGE” USING CONTENT-BASED ANALYSIS OF PROMOTIONAL PHOTOS

Jānis Lībeks , Douglas Turnbull

Swarthmore College
Swarthmore, PA 19081
{jlibeks1, turnbull}@cs.swarthmore.edu

ABSTRACT

We are interested in automatically calculating music similarity based on the visual appearance of artists. By collecting a large set of promotional photographs featuring artists and using a state-of-the-art image annotation system, we show that we can successfully annotate artists with a large set of (genre) tags. This suggests that we can learn some notion of artist similarity based on visual appearance. Such a similarity measure provides us with a novel *query-by-image* retrieval paradigm for music discovery.

1. INTRODUCTION

Long before Michael Jackson made music videos for MTV, and even before Elvis played The Ed Sullivan Show, the outward appearance, or *image*, of artists has played an important role in shaping how their music is received by audiences. Whether this image is carefully constructed by a public relations consultant or results from an unintentional lifestyle habit of the performer, it encodes valuable information that helps to place the artist into a musical context. For example, when seeing a group of four men take the stage wearing black t-shirts, studded black leather belts, tight black jeans, and long unkempt hair, it would be reasonable to expect them to play heavy metal.

To this end, we are interested in constructing a new measure of music similarity based on *visual* appearance. Such a measure is useful, for example, because it allows us to develop a novel music retrieval paradigm in which a user can discover new artists by specifying a query image. Images of artists also represent an unexplored source of music information that may be useful for semantic music annotation (e.g., associating *tags* with artists [10]).

In this paper, we describe a prototype computer vision system that can both compute artist similarity and annotate artists with a set of (genre) tags based on promotional photographs of the artists (see figure 1). The system is based on a state-of-art approach to content-based image annotation. The approach incorporates multiple forms of color and texture information and has been shown to outperform numerous alternative approaches on three standard image data sets. In order to use this approach for artist annotation, we modify it in a straight-forward manner so that we can use multiple images per artist to significantly improve performance.

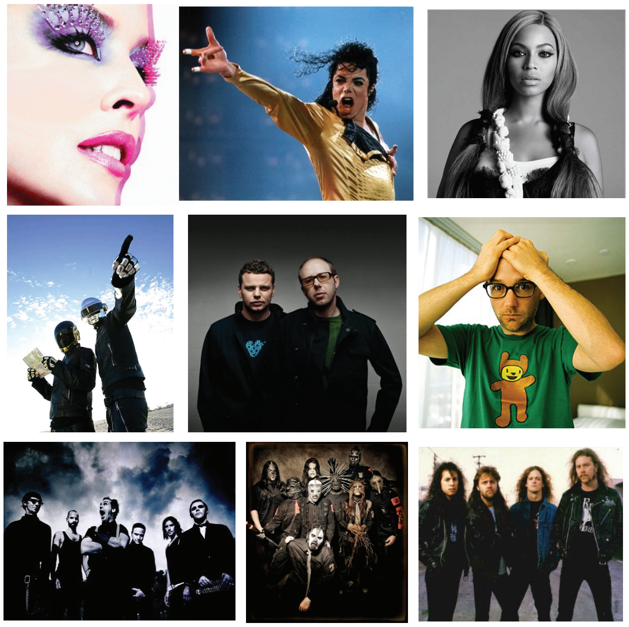

Figure 1. Example promotional photos of artists with the tags *pop* (1st row), *electronic* (2nd row), *metal* (3rd row).

2. RELATED WORK

Despite a thorough literature search, we have been unable to find previous work that uses computer vision to analyze promotional photos of music artists. However, computer vision has been used for various music-related tasks such as optical music recognition [1], identifying documents with music notation [2], and identifying lyrics in scores [3]. In addition, standard computer vision techniques have been applied to 2-D representations (e.g., spectrograms) of music for identification and fingerprinting [5].

Within the extensive computer vision literature, there are two general tasks that are related to our work. First, content-based image retrieval (CBIR) involves computing similarity between pairs of images. Deselaers et al. [4] provide a recent survey of CBIR research and describe a number of useful image features, many of which are used in this paper.

The second relevant task is image annotation. For this task, the goal is to annotate an image with a set of tags (e.g., “sky”, “polar bear”, “forest”). Makadia et al. [7] recently proposed a system that combines color and texture features using Joint Equal Contribution (JEC) as a baseline approach for this task. However, they unexpectedly found that this approach performs better than a number of (more complex) systems. We use JEC as the core of our artist annotation system but extend it to use multiple images of each artist.

3. IMAGE SIMILARITY

To compute image similarity between two images, we use the JEC approach that was recently developed by Makadia et al. [7]. The basic system involves computing seven separate distances between each pair of images. The seven distances are normalized and combined into one distance by taking the average over the seven distances.

3.1. Image Features

The first three distances are related to color information. For each image, we compute one color histogram over each of three color spaces: red-green-blue (RGB), hue-saturation-value (HSV), and LAB. The three color histograms are 3-dimensional histograms extracted on 16 equally spaced bins for each color channel. The interval of the bins is determined from the possible range of values for each channel in each of the respective color space. Each color histogram is represented as a $16^3 = 4096$ dimensional feature vector where each element of the vector represent the (normalized) count of pixels that fall into a color bin. As in Makadia et al., we calculate the L_1-distance when comparing two RGB or two HSV histograms, and calculate the KL-divergence when comparing two LAB histograms.

The other four distances are related to two types of texture information: Gabor and Haar features. For the Gabor features, a grayscale version of the image is convolved with complex Gabor wavelets at three scales and four orientations to create 12 response images. A histogram of response magnitudes is performed using 16 equally-spaced bins between experimentally-determined maxima values. Finally, the 12 histograms are concatenated, creating a final 192-dimensional feature vector. This representation is referred to as *Gabor* in this paper. A second Gabor feature, called *GaborQ*, is calculated by averaging the response angles of the 126x126 image over non-overlapping blocks of size 14x14, quantizing to 8 values and concatenating the rows of each of the twelve resulting 9x9 images, resulting in a 972-dimensional feature vector. We compute both Gabor and GaborQ distances for each pair of images by calcluating L_1-distance.

For the Haar features, we take three Haar filters at three different scales, and convolve them with a (downsampled) 16x16 pixel grayscale version of the image. The simple concatenation of the response image, a 2304-dimensional vector, was called *Haar*. A second quantized version, referred to as *HaarQ* is found by changing each image response value to 1, 0 or -1 if the initial response value is positive, zero, or negative, respectively, again, producing a 2304-dimensional vector. Again, we calculate the Haar and HaarQ distance by computing the L_1-distance between pairs of Haar and pairs of HaarQ vectors.

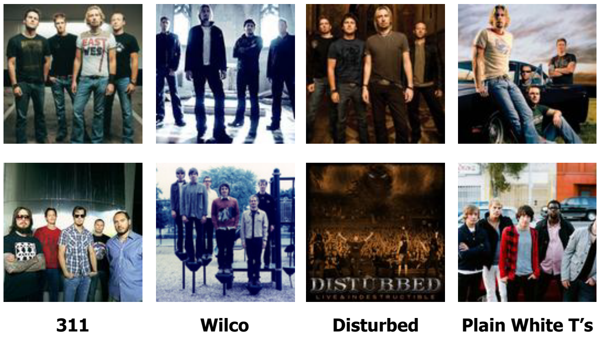

Figure 2. The four images for the artist Nickeclback on the top row, with the closest image for each on the second row.

Table 1. Image-based Artist Annotation. For a given seed artist (e.g., Nickelback), we retrieve the ground truth tag annotation vectors for the artists with the most similar images (e.g., 311, Wilco, ...) and then average their annotation vectors to calculate a predicted annotation vector.

	rock	indie	electronic	pop	punk	...
311	1	1	0	0	1	...
Wilco	1	1	0	0	0	...
Disturbed	1	0	0	0	0	...
Plain White T’s	1	1	0	1	1	...
Predicted Tags From JEC	1	0.75	0	0.25	0.50	...
Truth Tag For Nickelback	1	1	0	0	0	...

3.2. Joint Equal Contribution

To combine the distances of the seven features, we use Joint Equal Contribution (JEC). This is done by normalizing the distances for a feature by the maximum distance between any pair of images. This results in normalized distances in the range [0,1], where 0 denotes that the two images are the same, 1 denotes the most dissimilar pair of images. To combine feature vectors, we average the seven normalized distances over each pair of images.

4. ARTIST ANNOTATION EXPERIMENTS

To explore artist similarity, we consider the related problem of annotating images with genre tags [9]. That is, we assume that two artists are similar if they have been associated with a similar set of genres. The genre tags are provided by Last.fm[1] and are determined by a large number of individuals through a social tagging mechanism. Our system

[1] http://last.fm

Table 2. Performance of JEC Image Annotation on Artist Annotation Task.

Feature	AUC	MAP
Random	*.500*	*.099*
RGB	.564	.139
HSV	.573	.151
LAB	.571	.140
Gabor	.527	.111
GaborQ	.517	.111
Haar	.544	.122
HaarQ	.524	.115
JEC	**.585**	**.159**

works by first finding visually similar artists to a given seed artist, and then *propagating* genre labels from these artists to the seed artist [6]. We argue that if the true genre tags for the seed artists are related to the propagated tags for a seed artists, then our system is correctlly finding some notion of music similarity based solely on visual appearance.

4.1. Data

Using Last.fm, we collect a set of promotional images and a set of genre tags for a large number of artists. First, we create a vocabulary of genres by picking the 50 most popular genre tags on Last.fm[2]. Next, for each tag we gather a list of the 50 most representative artists, and for each artist we select all the tags from the artist that appeared in our vocabulary of tags[3]. This resulted in a set of 1710 unique artists and a (Boolean) tag matrix with an average of 4.74 tags per artist. Finally, for each artist, we attempt to download the 5 most popular promotional photos from Last.fm. Popularity is determined by Last.fm and appears to be related to the number of positive and negative votes that each image receives by their users. The downloaded images are precropped to 126x126 pixels by Last.fm. This resulted in a set of 8527 images, an average of 4.99 images per artist.

4.2. Tag Propagation

To evaluate the particular set of image similarity features, we compute a predicted tag vector for each artist. First, for each of the five images of the artist, we find the 1-nearest neighbor image from the set of all images (excluding other images of the artist) using the distances obtained using JEC with our image similarity features. Next, we average the tag annotation vector for each of the matched artists. Thus, for each artist we have a predicted tag vector of values in the range [0,1], with 1 meaning that all five visually similar artists are associated with the genre tag. See figure 2 and table 1 for an illustrative example of the annotation process.

Next, we compute two information retrieval (IR) performance metrics for each tag: Area under the ROC curve (AUC) and Mean Average Precision (MAP). For each tag, we starting by ranking artists by their predicted tag value for a given tag, and then calculating a performance metric using the ground truth tags for the artists. The ROC curve compares the rate of correct detections to false alarms at each point in the ranking. A perfect ranking (i.e., all the relevant artists at the top) results in an AUC equal to 1.0. We expect the AUC to be 0.5 if we randomly rank artists. Average Precision (AP) is found by moving down our ranked list of test artists and averaging the precisions at every point where we correctly identify a relevant artist. More details on these standard IR metrics can be found in Chapter 8 of [8]. To evaluate a selection of image similarity features, we compare the averages of the AUC and MAP over all 50 genre tags.

5. RESULTS

First, we compare the set of image features to determine which features are most appropriate to the problem. Next, we indicate the performance of individual tags in our dataset, given the best combination of features.

5.1. Evaluation of features

We follow the evaluation pattern used by Makadia et al. [7] by first considering the performance of each image similarity feature seperately, and then looking at the performance when we combine features.

The results for the seven image similarity features as well as the combined JEC approach are listed in table 2. All of the features perform significantly better than random[4]. For this data set, color features work best (RGB, HSV, LAB). More importantly, we note that the combination of all seven features using JEC performs significantly better than any individual feature. In addition, we explored removing individual features before computing JEC, without a significant impact on performance.

Table 3. Effect of using multiple images of each artists (using AUC).

# of images	1	2	3	4	5
JEC w/o Haar	0.517	0.542	0.567	**0.585**	0.566

As shown in table 3, we find that increasing the number of images used for each artist improves performance, but only up to the fourth image; adding the fifth most popular image of the artist decreases performance. Thus, we used the four most popular promotional photos for each artist for the rest of our analysis.

5.2. Tag performance

In table 4, we list the AUC performance of individual tags with JEC. Some of the tags, such as *metal*-related, *dance*,

[2] Tags such as ‘seen live’ and ‘vocalists’ were discarded, since they do not fit the notion of a music genre.

[3] To avoid spurious genre labels for an artist, we retain only tags that have a Last.fm score of 5 or more on a 100-point scale for that artist.

[4] For comparing image features, statistical significance is determined using a two-tailed paired t-test over the $n = 50$ tags with $\alpha = 0.05$.

Table 4. AUC performance of each individual tag using JEC. Performance that is not significantly better than random is indicated with *italic* font.

Tag	AUC
Random	*0.50*
melodic death metal	**0.70**
metal	0.69
power metal	0.68
death metal	0.67
metalcore	0.66
heavy metal	0.66
dance	0.64
classical	0.63
indie pop	0.63
thrash metal	0.63
pop	0.62
rnb	0.62
trip-hop	0.62
electronic	0.62
electronica	0.62
black metal	0.61
ambient	0.61
indie	0.61
trance	0.61
indie rock	0.59
hardcore	0.59
chillout	0.58
emo	0.58
industrial	0.58
rock	0.58
jazz	0.58
alternative	0.58
alternative rock	0.58
new wave	0.57
electro	0.57
progressive metal	0.56
post-rock	0.56
hip-hop	0.56
house	0.56
techno	0.56
hard rock	0.55
progressive rock	*0.55*
hip hop	0.55
reggae	*0.55*
punk rock	*0.55*
soul	0.54
punk	0.54
experimental	0.54
blues	*0.53*
rap	*0.52*
classic rock	*0.52*
psychedelic	*0.52*
folk	*0.50*
country	*0.50*
funk	*0.47*

classical, and *indie pop*, tend to perform best (i.e., AUC > 0.62). It is also worth noting that the first four most successful tags contain the word *metal* (specifically, *melodic death metal*, *metal*, *power metal*, and *death metal*), indicating that the top-level genre *metal* has a specific visual appearance that makes it easy to identify, based on our set of features.

On the other hand, the image annotation performance for 10 of our 50 tags (e.g., *psychedelic*, *folk*, *country*, *funk*) are statistically no different than random[5]. This does not mean that there is no useful information in these images, but rather that our image annotation system does not adequately extract, encode or model this information. For example, one can imagine a computer vision system that can explicitly detect cowboy boots and 10-gallon hats in order to identify *country* artists.

6. DISCUSSION

In this paper, we have shown that we can automatically annotate artist with a large set of genre tags using the images of the artist. We believe that this is an exciting new research direction because it provides us with a novel *query-by-image* music discovery paradigm. To this end, we have developed a prototype web-based music image browser for exploring music similarity and annotation called the *Artist Image Browser*[6].

We note that the system presented in this paper is only a *baseline* approach in that we have only considered low-level features, such as color and texture, but paid no attention to higher-level features (e.g., object detection). To this end, future work will include face, body and object detection. Future work will also explore human-level performance on the artist annotation task. That is, we are interested in better understanding how humans interpret images of both known and unknown artist, and whether this affects their experience when listening to music.

7. REFERENCES

[1] D. Bainbridge and T. Bell. The challenge of optical music recognition. *Computers and the Humanities*, 2001.

[2] D. Bainbridge and T. Bell. Identifying music documents in a collection of images. In *ISMIR*, pages 47–52, 2006.

[3] J. A. Burgoyne, Y. Ouyang, T. Himmelman, J. Devaney, L. Pugin, and I. Fujinaga. Lyric extraction and recognition on digital images of early music sources. *ISMIR*, 2009.

[4] T. Deselaers, D. Keysers, and H. Ney. Features of image retrieval: An experimental comparison. *Information Retrieval*, 2008.

[5] Y. Ke, D. Hoiem, and R. Sukthankar. Computer vision for music identification. In *CVPR*, June 2005.

[6] J.H. Kim, B. Tomasik, and D. Turnbull. Using artist similarity to propagate semantic information. *ISMIR*, 2009.

[7] A. Makadia, F. Pavlovic, and S. Kumar. A new baseline for image annotation. *ECCV*, 2008.

[8] C.D. Manning, P. Raghavan, and H. Schtze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[9] C. McKay and I. Fujinaga. Musical genre classification: Is it worth pursuing and how can it be improved? *ISMIR*, 2006.

[10] D. Turnbull, L. Barrington, and G. Lanckriet. Five approaches to collecting tags for music. *ISMIR*, 2008.

[5] For each tag, we determine statistical significance ($\alpha = 0.05$) by comparing the AUC for the tag with $n = 1000$ bootstrapped estimates for AUC values based on different random rankings of the artists. This allows us to directly calculate a p-value from the empirical distribution of AUC values.

[6] Artist Image Browser: http://www.cs.swarthmore.edu/aib/