Choosing which Clothes to Wear Confidently: A Tool for Pattern Matching

Nektarios Paisios
Department of Computer Science, New York University
251 Mercer St.
New York, NY 10012 USA
nektar@nyu.edu

Lakshminarayanan Subramanian
Department of Computer Science, New York University
251 Mercer St.
New York, NY 10012 USA
lakshmi@cs.nyu.edu

Alexander Rubinsteyn
Department of Computer Science, New York University
251 Mercer St.
New York, NY 10012 USA
alexr@cs.nyu.edu

ABSTRACT

This work attempts to make a first step in computationally determining whether a pair of clothes, in this case of a tie and a shirt, can be worn together or not, based on the current social norms of color-matching. Our aim is to give visually impaired persons the ability, using snapshots taken by their mobile phones, to independently and confidently be able to choose from their wardrobe which set of clothes they can wear together.

Given a sample of 41 pairs of shirts and corresponding matching ties of 74 pixels squared and using a color-histogram of 8 bins, we show that with Ridge Regression we can achieve a 10-fold cross-validation classification error of 0.175, with a standard neural network with 10 hidden neurons a 10-fold cross-validation classification error of 0.05 and with a Siamese neural network an accuracy of only 0.33.

Keywords

Camera Phone, Clothes Matching, Visually Impaired.

Categories and Subject Descriptors

K.4.2 [**Computers and Society**]: Social Issues - Assistive Technologies for Persons with Disabilities; H.5.2 [**User Interfaces**]: User Centered Design, Prototyping.

General Terms

Algorithms, Design, Experimentation

1. INTRODUCTION

Being dressed in a socially acceptable combination of clothes is extremely important in a modern society, especially in cases where professionalism is synonymous to attire. Wearing clothes that match in color with one another is, up to a point, both a skill and a part of common sense. Visually impaired persons, however, have difficulty matching their clothes as they cannot readily identify their color. Even in cases where other means of color identification are available, such as electronic color detectors and marking clothes with a tactile tag or other material, blind people who had no sense of color before in their lives, have no way of knowing if a set of two or more color-combinations on their different pieces of clothes match or not. Training oneself to memorize which colors match or which do not can certainly be a solution, but it is certainly hard to be achieved fully given the vast number of color-combinations, and all the more so in the case of someone who has never physically experienced them. Previous work has made use of very specific color or pattern identification techniques [3, 2], which might be fragile in the face of the huge variety in clothes design. This work attempts to evaluate 3 standard learning methods, a Ridge Regression, a standard neural network and a Siamese Neural Network to identify if a sample of shirt/tie pairs match or not.

2. CLOTHES MATCHING PROBLEM

The basic problem we aim to address is: given two images corresponding to a pair of clothes, we need to determine if the pair of clothes match or not. While there may be several aesthetics espoused by different individuals, we take a simplistic approach in this problem.

Consider a visually impaired user who uses his/her specific fashion designer to manually suggest whether a pair of clothes match or not. For each visually impaired user, we construct a training sample set that comprises of matching clothes marked by a specialist. Given this training set, can we develop a learning algorithm to automatically infer if two new pair of clothes match or not?

In this paper, we consider the example problem of matching shirts and ties; specifically because ties may be associated with visual patterns making the problem harder than a simple color matching problem. This is because in the simplest case we could have just created a simple color matching rulebook. Our research question is: Given a training set consisting of both matching and non-matching pairs of shirts and ties, can we develop an automatic algorithm to determine if a new pair of shirt and tie match or not? We outline our methodology and learning algorithm in the next section.

Figure 1: A matching pair of shirts and ties

Shirt Matching Tie

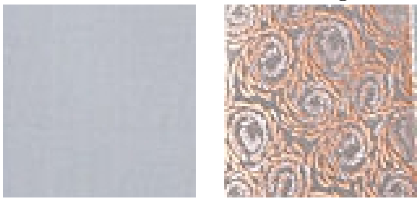
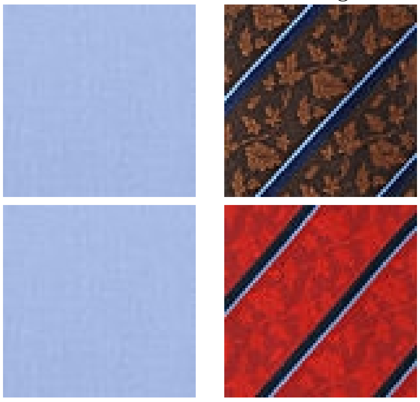


Figure 2: The same shirt matches with more than one tie

Shirt Matching Tie



3. METHODOLOGY

3.1 Sampling

A total of 41 pairs of shirts and corresponding matching ties were used with sizes of 74 by 74 pixels. Each image was hand-picked so that for the shirt the part shown is the part closest to the place where a tie would be worn, i.e. no unrelated parts of a shirt, like the sleeves, were included. For our sample set, each pair was included together with a label indicating that they match. Also, non-matching pairs were artificially created by pairing each shirt and each tie with itself, creating a total of 82 non-matching pairs. In total there were 123 samples in our training/testing set. However, other pairings were also tried, such as the all possible pairings. However, for the results reported here we chose to go with the simpler approach, as our pairs labeled ”non-matching” are certainly so, a fact which cannot be reliably claimed for the non-matching pairs of the all pairs set. This is because many ties can match with one shirt, making some of the ”non-matching” labels in the all-pairs set invalid.

3.2 Data Preparation

Each image was loaded and used to create a color histogram. The histogram was created by dividing the 3-dimensional color space of red, green and blue into a configurable number of equal bins (we tried experimenting with 8*8*8, 4*4*4 and 2*2*2 bins) and by determining in which bin each pixel falls. This was carried out after the luminance of each image was factored out by normalizing it. Finally, the sum of pixels at each bin was normalized creating a normalized color histogram, giving a number between 0 and 1 for each feature in the features vector in order to achieve better performance in the learning algorithms, although unnormalized histograms were also tried with worse performance. The color histograms of each matching pair and non-matching pair, chosen above, were then concatenated into one feature vector and a corresponding label of +1 and 0 was attached.

Figure 3: Non-matching pairs of shirts and ties

Shirt Non-Matching Tie

3.3 Learning Algorithms

The following learning approaches were tried:

- Ridge Regression: This algorithm attempts to calculate the vector of weights which would return for each seen data point its corresponding score. In addition, it tries to minimize the norm of the weight vector and so it is called Ridge Regression to distinguish it from linear regression. Although regression is not a classification algorithm, it can be used to derive results on unseen data which can then be compared to an artificial threshold to derive the class. In our case, we set the threshold at 0.5 and so if regression returns a value which is less than 0.5 it receives the 0 (not a match) label and if greater than 0.5 it receives the 1 (a match) label.

- Standard neural network: A network with two hidden layers is trained using either stochastic or batch training, using the Sigmoid as its activation function.

- Siamese Neural Network: The Siamese Neural Network which is described in [1], is implemented as follows:

  1. Two sets of outputs for each layer are stored each one corresponding to one of the two samples in the given training pair.
  2. The loss function is set to the derivative of a hinge loss function, which aims to make the distance between outputs smaller when the pair of images matches but which tries to make the distance large, but only up to a threshold, when they do not.
  3. Uses the same error for the output layer for both images, but reverses the sign of the error for the second image.

Table 1: Performance of learning algorithms

Algorithm	Regression Error	Classification Error
Regression	0.704427	0.175
Neural	0.0955543	0.05

Table 2: Type-1 and Type-2 errors of learning algorithms

Algorithm	Type1 Error	Type2 Error
Regression	0.3	0.11253
Neural	0.125	0.0125

4. RESULTS

Table 4 and Table 4 show the regression and the 10-fold cross-validated classification errors for two of our three approaches (regression and standard) neural network, together with the type one and type two errors, i.e. false-positives over (true-positives + false-positives) for the type one error and false-negatives over (true-negatives + false-negatives) for the type two error. The Siamese Neural Network was tried with 10 output and 10 and 100 hidden neurons but as expected produced very bad results, (classification error = $\frac{2}{3}$). This is because, as our inputs are structured, the non-matching pairs are made up of two images that are the same in order to give the standard network the most negative samples that it can get. However, it is impossible for a Siamese Neural Network to identify the distance between two identical images, as by definition of this algorithm the distance should be 0. More experimentation is clearly needed by providing pairs which visually do not match but this is left for a future work. In addition, the matchings are not symmetric, meaning that a shirt might match with many ties but not the other way round. A suitable solution perhaps was to add an extra dimension to the input to identify if a sample is a tie or a shirt. Ten hidden neurons were also found sufficient for the standard neural network in addition to a learning rate value of 0.1, with a binning of 4 for the color histogram as other values did not change the 10-fold cross-validation regression error substantially. The stopping condition for the cross-validation was 0.00001. The regression error is calculated by taking the forbeenius norm of the difference between the expected and actual outputs for each data point and averaging over all points and over all cross-validation folds.

5. CONCLUSIONS

This paper defines a new and potentially interesting problem to the HCI community on the clothes matching problem and its relevance in the context of visually impaired users. This problem was actually motivated by the real-world experiences of a visually impaired user who has been the primary lead researcher of this work. We have presented an early version of a simple algorithm to tackle this problem especially in the context of shirts and ties.

Based on our results, we note that the standard neural network has better accuracy as it performs better than Ridge Regression on our test samples. However, more work is needed, especially in finding more samples such as for other types of clothes, to be able to deal with the problem of clothes-matching more effectively. In addition, symmetric matching pairs should be found or artificially created in order to be able to deploy the already designed Siamese Neural Network. More importantly, our algorithm should be enhanced to take into consideration other characteristics of the clothes, such as their texture or their design patterns. The above will necessitate a user study to discover how humans actually distinguish between sets of clothes that match and sets that do not.

6. REFERENCES

[1] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *Computer vision and pattern recognition, 2006 IEEE computer society conference on*, volume 2, pages 1735–1742. IEEE, 2006.

[2] J. Rose. Closet buddy: dressing the visually impaired. In *Proceedings of the 44th annual Southeast regional conference*, pages 611–615. ACM, 2006.

[3] S. Yuan. A system of clothes matching for visually impaired persons. In *Proceedings of the 12th international ACM SIGACCESS conference on Computers and accessibility*, pages 303–304. ACM, 2010.