Visual Recognition of Paper Analytical Device Images for Detection of Falsified Pharmaceuticals

Sandipan Banerjee[1], James Sweet[1], Christopher Sweet[1], and Marya Lieberman[2]
[1]Department of Computer Science and Engineering,
[2]Department of Chemistry and Biochemistry,
University of Notre Dame
{sbanerj1, jsweet, csweet1, mlieberm}@nd.edu

Abstract

Falsification of medicines is a big problem in many developing countries, where technological infrastructure is inadequate to detect these harmful products. We have developed a set of inexpensive paper cards, called Paper Analytical Devices (PADs), which can efficiently classify drugs based on their chemical composition, as a potential solution to the problem. These cards have different reagents embedded in them which produce a set of distinctive color descriptors upon reacting with the chemical compounds that constitute pharmaceutical dosage forms. If a falsified version of the medicine lacks the active ingredient or includes substitute fillers, the difference in color is perceivable by humans. However, reading the cards with accuracy takes training and practice, which may hamper their scaling and implementation in low resource settings. To deal with this, we have developed an automatic visual recognition system to read the results from the PAD images. At first, the optimal set of reagents was found by running singular value decomposition on the intensity values of the color tones in the card images. A dataset of cards embedded with these reagents is produced to generate the most distinctive results for a set of 26 different active pharmaceutical ingredients (APIs) and excipients. Then, we train two popular convolutional neural network (CNN) models, with the card images. We also extract some "hand-crafted" features from the images and train a nearest neighbor classifier and a non-linear support vector machine with them. On testing, higher-level features performed much better in accurately classifying the PAD images, with the CNN models reaching the highest average accuracy of over 94%.

1. Introduction

Billions of people in developing countries are facing the problem of fake or substandard pharmaceuticals. Poor qual-
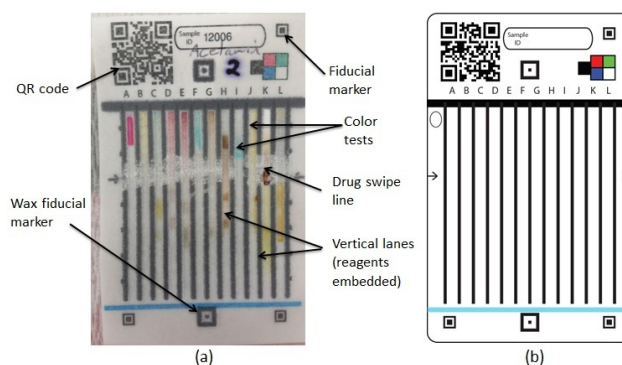

Figure 1: (a) Unrectified PAD image of the drug *acetaminophen* with different reagents in the 12 different vertical lanes, (b) PAD image artwork.

ity medications may contain harmful or inert ingredients that fail to treat the patient's underlying condition. Drugs which contain sub-therapeutic levels of an active ingredient contribute to development of drug-resistant pathogens. The human toll of these products is large. Falsified antimalarials have been estimated to kill 120,000 children under the age of 5 each year [1]. Although there are pharmacopeia methods for analysis of medications, they often cannot be applied in the developing world due to lack of technical and regulatory resources, and studies consistently show that 15 - 30% of medications for sale in Africa, southeast Asia, and other low resource settings are substandard or fake. To address this problem Weaver et al. proposed the idea of using inexpensive (under $1) paper analytical devices (PADs) that contain libraries of color tests to respond to the chemicals inside a pill [3]. The PADs have 12 vertical "lanes" in which chemical reagents are embedded. When a drug is swiped across these lanes and the PAD is dipped in water, the reagents react with the different compounds in the drug, generating characteristic color tones in the vertical
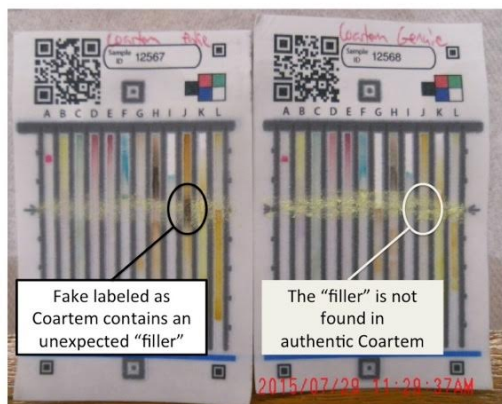
Figure 2: PAD images with fake and authentic versions of the drug *coartem*.

Figure 3: The different phases of the color testing process of a PAD.

lanes, usually at or above the drug swipe (see Figure 1.a). The distinctive set of color tones generated for a particular drug acts like its "fingerprint", and is considerably different from that of a counterfeit version of the same drug (Figure 2).

The different steps of the color testing process of a PAD has been illustrated in Figure 3. At present, the user evaluates the results of the test by visually comparing their PAD to stored images of cards run with standard pharmaceutical samples. However, reading the card results is much more challenging than actually running the chemical tests with them. Important differences in the color tones are not always picked up by human users, and the time and effort required to train people to read the cards correctly and verify their capability is a barrier to wider scaling of the use of the cards. To tackle this, we have developed a visual recognition system which can classify a PAD image based on the pharmaceutical drug being tested. Only 18% of people in Africa had access to the internet in 2013 although two out of three used a mobile phone [2], we thus decided that the image recognition system should be compatible with the cell phone network. The user should be able to submit a PAD image through her cell phone and the system should return a quality score of the tested drug back to the user. By using messaging services instead of the internet to transfer card images and the results, this system is designed to be usable by people who live in developing countries. In this paper, we discuss the different steps of the visual recognition process of the PAD images. Different feature extraction methods and their performance in recognizing the drugs using different classifiers are also presented.

The remainder of this paper is structured as follows. Prior research relevant to this problem, and how our work is different from them, is discussed in section 2. In section 3 we describe the images we prepared for this work
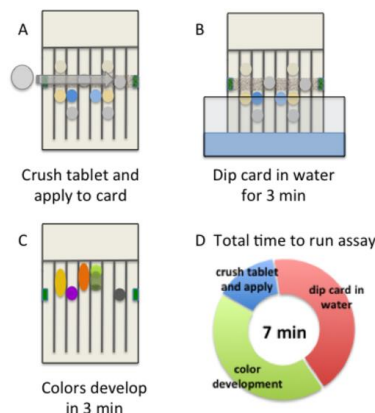
and the methods used to decide on a design choice for the PAD cards. Section 4 explains the different feature extraction methods and the classifiers used for the experiments. The experimental results are discussed and further analyzed in section 5. Section 6 concludes the paper.

2. Related work

Modern medicine has reached great heights in the last one or two decades and it has brought about a huge growth of pharmaceuticals in the market. However, the police, regulators, healthcare providers, and patients are often faced with the task of identifying a mystery pill or tablet. Much of the work that has been done on automatic drug identification is from the last decade. Most of the researchers have discussed automatic pill recognition protocols based on the color, shape and imprint information on the pills in [4, 5, 6]. The work of Lee *et al.* in [7] is worth mentioning in this regard, where they try to retrieve images of similar pills when provided with a query pill image. They extract the color histogram and the Hu moments [24] based on the shape of the pill as features. Additionally, SIFT [25] and MLBP [26] descriptors are used to encode any imprint information present on the pills. The matching accuracy for these pills is computed using both individual features and their different combinations. They report a rank-1 retrieval accuracy of about 73%. Such pill identification systems, such as [8, 9, 10], are available online for general purpose use.

Our work is considerably different from what has been proposed in these papers as we crush the pills and swipe them on the PAD cards for classification based on their chemical composition. Thus, instead of identifying the pill based on its appearance we recognize the drug based on its chemical composition, which is revealed by the chemical "fingerprint" it makes in reacting with the reagents embedded in the PAD cards. The main objective of our work is to

detect pills whose formulations do not match the authentic version. Fake pills can match the real ones exactly in appearance, but if the ingredients of the pill are different, the color fingerprint generated on the PAD will be considerably different from that of the real drug. An additional advantage of the PAD method is that some types of falsification can be detected even if an authentic sample of the particular brand of pill is not available. To our knowledge, there is no published work in the literature which uses vision techniques for the recognition of the PADs and our work is the first of its kind.

3. Preparation of the PAD cards and their images

For correct identification of different pharmaceuticals using the PAD images, the cards have to be first prepared with the best set of up to 12 reagents (one for each lane) which produces the most unique color tones for the different drugs. For our experiments, 26 pharmaceutical drug samples[1] were chosen either because they have been reported in the literature as ingredients in falsified formulations (*e.g.* corn starch, acetaminophen) or because they are in the list of WHO essential medicines [11], with high sales volume in the developing world (*e.g.* amoxicillin, isoniazid). For this 26 drug samples, we choose the best 12 from a set of 24 reagents. In order to analyze the card images, the raw unaligned image taken with a cell phone needs to be rectified and properly aligned prior to the feature extraction process. Therefore, the card preparation and image rectification are two key steps in proper drug classification. We describe the PAD image alignment process first, and then the optimal reagent selection process below.

3.1. PAD image rectification

The PAD cards are photographed after the reactions between the drug and each reagent have taken place. However, the lanes in the resulting image are unaligned and don't match the locations of the original PAD artwork (Figure 1.b). So, some transformation is required before the analysis can proceed. The images have varying resolution, scaling, rotation, offsets and perspective attributes to correct for. The method we use to accomplish this is described below.

The PAD card is pre-printed with a QR code in the upper left hand corner (containing the image metadata) and three additional fiducial marks at the remaining corners. The PAD processing software uses a variant on the QR

---

[1]The drugs are: *acetaminophen*, *acetylsalicylic acid*, *amodiaquine*, *amoxicillin*, *ampicillin*, *artesunate*, *azithromycin*, *calcium carbonate*, *chloramphenicol*, *chloroquine*, *ciprofloxacin*, *corn starch*, *DI water*, *diethylcarbamazine*, *dried wheat starch*, *ethambutol*, *isoniazid*, *penicillin G*, *potato starch*, *primaquine*, *quinine*, *rifampicin*, *streptomycin*, *sulfadoxine*, *talc* and *tetracycline*.
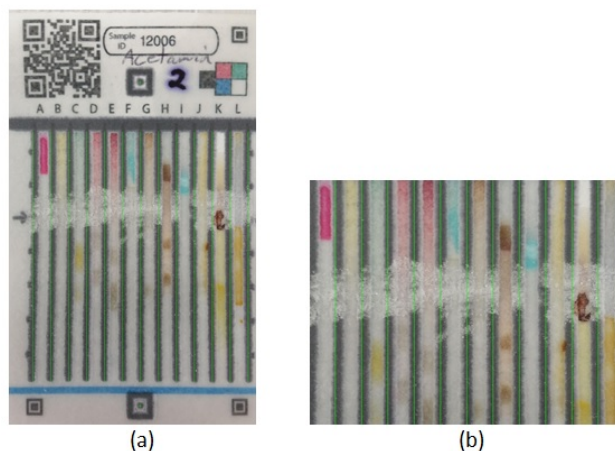

Figure 4: (a) The rectified version of the PAD image from Figure 1, (b) the cropped out salient region from the rectified PAD image.

code alignment marker finding algorithm to locate the fiducial marks, together with the QR code markers, to acquire a maximum of six reference points. The resulting points are processed with the popular OpenCV [12] library to do a offset, scaling, rotation and perspective correction to match the coordinates of the artwork. In addition to the ink print layer, the face of the card is printed with wax lines that are baked into the paper to separate the reagent lanes. Since the lanes can be misaligned with the original ink layer due to the printing process, two wax fiducial markers are added and a template matching scheme within OpenCV is used to find their coordinates. These coordinates are used to remove the offset and rotation of the lanes within the rectified version of the PAD image, example shown in Figure 4.a (730x1220 in size).

3.2. Selection of the optimal set of reagents

24 different reagents were tested to determine which ones produced the most useful color outcomes for classification of the 26 pharmaceuticals and fillers tested in this study. We prepared a set of cards in which 9 of the 12 lanes contained a given reagent; the lanes were arranged in groups of three with a spacer between each group (Figure 5). One of 26 pharmaceuticals or fillers was swiped across these lanes to give groups of light, medium and heavy concentrations of the analyte.

The cards were activated by dipping the bottom edge in water, as shown in Figure 3. The different reagent/analyte combinations produce a variety of colors, and often a single lane gives multiple colors in different locations, which can be difficult for a human to interpret. Because the chemical reactions are designed to generate strong colors, we consider the color "blobs" having the color intensity farthest from white (the background card color) to be the most
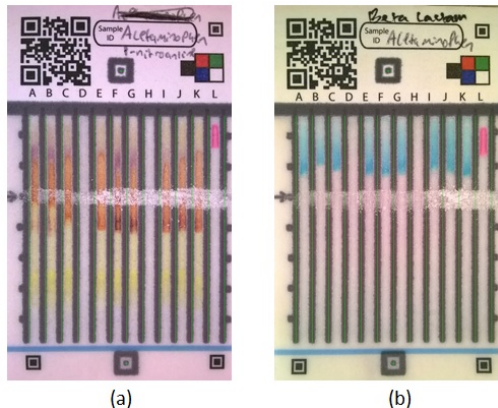
Figure 5: (a) Rectified PAD image of the drug acetaminophen with the reagent that detects phenol groups in the 9 lanes from A - K, (b) rectified PAD image of acetaminophen with the reagent that detects beta lactam groups in the 9 lanes from A-K. The drug concentration increases from the left to the right of image, the pink hue in lane L is from the timer agent.
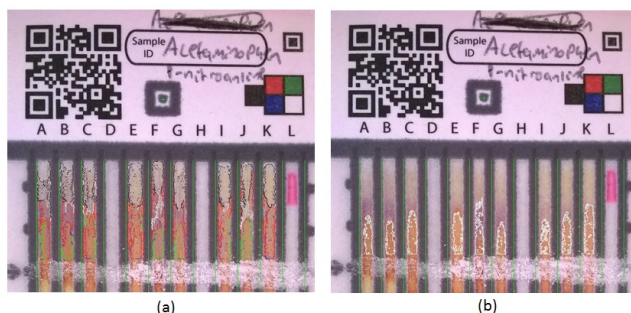

Figure 6: (a) The overlapping blob contours generated by the region growing algorithm in each lane, (b) the estimated actual reaction blob contour in each lane.

meaningful color result of that lane (the reaction blob). In order to locate it on the image, we first divide the lane space above the swipe line into five equal regions. Then we run a region growing algorithm from the centroid of each region, and compute the 5 connected components from them. Since the regions are close to each other and there are at most 1 or 2 residual blobs, multiple connected components are from the same blob and they overlap, as can be seen in Figure 6.a.

We merge two such regions together if the number of overlapping common pixels between the regions is greater than a threshold. Experiments have shown the overlapping regions to merge best when this threshold is set as 0.35 times the size (# of pixels) of the bigger region of the two. Once this is done, we end up with 2 or 3 blobs in total per lane, one of which is the actual reaction blob and the others are residual. Since the residual blobs are generally fainter in color intensity than the reaction blob, we compute the

maximum difference across the R, G and B channels of the mean pixel intensity for each blob as shown in equation (1). We keep the blob with the highest value of *maxDiff* per lane as it is the farthest from the background card color (white), as shown in Figure 6.b.

$$maxDiff = max(\mid R_{mean} - G_{mean} \mid, \mid G_{mean} - B_{mean} \mid,$$
$$\mid B_{mean} - R_{mean} \mid)$$
$$(1)$$

9 reaction blobs are obtained in this way in the 9 lanes of the PAD image. The mean RGB intensity of these 9 blobs serve as a reaction color descriptor for a particular drug and a reagent. A database of such descriptors is built for all the possible pairs from the 26 drugs and 24 reagents. We investigate the uniqueness of each point in the database for a particular drug using the singular value decomposition (SVD) method. A 26x24 matrix (*M*) is built which contains the Euclidean distance of the mean RGB intensities between the drug and each reagent. Decomposing *M* by applying SVD, shown in equation (2), we obtain the singular matrices *U* and *V*, and the diagonal matrix *S*.

$$M = USV^t \qquad (2)$$

V is a 24x24 matrix which contains information about the contribution of each reagent to the singular values of *M*. Sorting the columns of *V* in a descending order of corresponding singular value lists the indices of the reagents by their uniqueness for the particular test drug. They can be termed as the optimal reagents for that drug. We capture the first index from the sorted list, *i.e.* the index of the reagent that produces the most distinctive reaction color with that drug, for all the 26 drug samples in a list. The list contains 9 different reagents, as many reagents react very distinctively with multiple drugs. Based on this list, we compute the norm between all drug sample pairs from information in matrix *M* to verify the uniqueness of the results. All samples are found to have an inter-class norm of more than at least 1 intra-class standard deviation of that drug. We added 2 more reagents which are known to produce unique results from their chemistry to the list[2]. We then prepared new 12-lane cards with a timer lane and the 11 optimal reagents, as shown in Figure 1.a, and used these cards to generate color fingerprints from the 26 active pharmaceutical ingredients and excipients. The colors from the chemical reactions between the drug and the reagents are formed either at or above the swipe line, so we crop out these salient regions from the rectified PAD images (resolution of 636x490), and

---

[2]Optimal reagents (lane A to lane L): Ni/nioxime timer, ninhydrin test for primary amines, biuret reagent, acidic cobalt thiocyanate, neutral cobalt thiocyanate, copper test for beta lactam, sodium nitroprusside, napthaquinone sulfonate, copper test for ethylenediamines, iodine test for starch, phenol test and ferric ion.
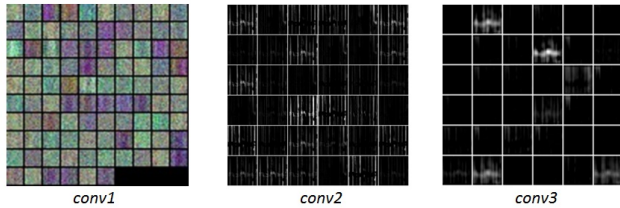
Figure 7: The visual representation of the filter and their responses from the 1st, 2nd and 3rd convolutional layers (*conv1*, *conv2* and *conv3* respectively) in Caffenet.

store them in the database, discarding the rest of each image (Figure 4.b).

4. Feature extraction and classification

Since the PAD images are not just a set of color tones but also contain characteristic blob patterns, we explored a broad set of features, ranging from low-level to high-level. Each feature used in our experiments and the classifiers used for it are described below.

L*a*b* color histogram: The primary distinguishing factor for classifying two PAD images of two different drugs is by comparing their color fingerprints. Some of the images have reaction blobs restricted to a certain number of distinguishable colors while others span across a wide variety of color tones. We capture a 90-dimensional binning of the L*a*b* histogram as a feature that characterizes the color distribution in the images.

GIST descriptors: We use the GIST descriptors (512 dimensions) as another low-level feature as it describes the spatial envelope of the image and provides a good representation of the visual field [23].

Color bank: This is the color bank feature proposed in [17]. The PAD image is first converted to a set of color names [18, 19] and the histogram of these names is extracted from dense overlapping patches of different sizes from the images. We learn a dictionary (of size 20) using k-means [20] from a random sampling of the patch histograms and apply locality-constrained linear coding (LLC) [21] to soft-encode each patch to dictionary entries. Then, max pooling is applied with a spatial pyramid [22] on the dictionary to obtain the final feature vector (420 dimensions).

Dense SIFT: The SIFT descriptor [25] captures the local patterns in the image in a scale and orientation invariant manner. We extract the SIFT descriptors from a dense grid over the images at different patch sizes. A dictionary is obtained by applying k-means (k = 256) on these descriptors and we soft encode the dictionary entries with LLC. Then the final feature vector (5376 dimensions) is obtained by applying max pooling with a spatial pyramid on the dictionary.

Convolutional neural network: The recent resurgence of deep learning models and their high performance in visual recognition tasks shows a lot of promise in this regard. Instead of hand-designed features traditionally used for visual recognition, the convolutional neural network (CNN) models extract features automatically in their intermediate layers from the input images while training the network. The fact that CNN based models have been winning the last few editions of the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [13], which deals with recognizing objects from a million images, underpins its robustness and the reliability of its automatic feature design process. We use Caffe [16] for feature extraction, training and testing two popular model architectures with our data, as described below.

- **Caffenet**: It is a clone of the network (5 convolutional + 3 pooling layers) which won the ILSVRC 2012 contest proposed by Krizhevsky *et al*. in [14] for general object recognition. We use only the architecture as it is, except in the last fully connected layer where the number of output is changed to 26, and train the network with our PAD image data from scratch. The features, both low and high level, are automatically captured by the network in the filter maps of its intermediate layers, as shown in Figure 7.

- **GoogLeNet**: It is the ILSVRC 2014 winning "very deep" 22-layer network architecture from Google, described in [15]. We use the architecture only, with the configuration of the fully connected layers changed, and train it with the PAD images from scratch. The features are captured in the intermediate layer filters.

To train and test the features discussed above, except those from the CNN models, we use two classifiers: a simple squared-Euclidean distance based nearest neighbor (kNN, where k = 1) and a multi-class support vector machine (SVM) with a radial basis kernel. The SVM training parameters are estimated using cross-validation on the dataset. We chose the softmax regression layer used for classification in the original networks described above, for training and testing with the CNN features. Thus, we use the entire pipeline of these models for both feature extraction and classification of the PAD images.

5. Experiments and results

For our experiments we prepare a dataset of 30 images (cropped salient regions) per drug, making a total of 780 images for the 26 drugs. We randomly select 20 images for training for each drug and use the remaining 10 for testing, giving us a total of 520 images for training and 260 for testing. The classifiers (kNN and SVM) are trained using the features described in the previous section. The CNN models

Table 1: Average classification accuracy of PAD images

Method	Accuracy (%)
L*a*b* histogram with kNN	138/260 (53.07%)
L*a*b* histogram with SVM	172/260 (66.15%)
GIST with kNN	216/260 (83.07%)
GIST with SVM	230/260 (88.46%)
Color bank with kNN	235/260 (90.38%)
Color bank with SVM	231/260 (88.84%)
Dense SIFT with kNN	200/260 (76.92%)
Dense SIFT with SVM	233/260 (89.61%)
Color bank + dense SIFT with kNN	238/260 (91.53%)
Color bank + dense SIFT with SVM	240/260 (92.30%)
Caffenet	**245/260 (94.23%)**
GoogLeNet	243/260 (93.46%)

Table 2: Average classification accuracy of non-structured PAD images with Caffenet

Method	Accuracy (%)
Unrectified images with Caffenet	233/260 (89.61%)
Images with random lane order with Caffenet	87/260 (33.46%)

train with the features they capture from the images automatically. We set a maximum cap of 100,000 iterations for training for both the models as our dataset is much smaller compared to ImageNet [27]. The network models tend to converge earlier than that, generally between the 70,000 - 80,000 iterations. The weight configuration (model state) which gives the best performance on the validation images is selected for testing. Since our data set is small in size, to handle overfitting (especially for the CNN models) on the training data, we perform training from scratch using a 3-fold cross validation approach on the available dataset and calculate the classification accuracy by averaging results from each experiment. The top-1 accuracy, averaged over the 3 folds, of each method in correctly classifying the test PAD images can be seen in Table 1.

Clearly, some methods are better than others in classifying the cards. For example, with the color histogram alone as the feature almost half the test images are misclassified while the predictions are much more accurate when using higher level features. However, no one method can be marked as the outright winner in a statistical sense, as their confidence intervals overlap over the 3 folds. Interestingly, both the CNN models do a good job in classifying the PAD images, with Caffenet churning out the most accurate average predictions for our dataset. The average confidence for each drug class that Caffenet makes a prediction for can also be found in Figure 8. Although the data is limited, still the network is quite precise in making the accurate predictions in majority of the cases considering that chance prediction confidence is 0.038 (1/26). Caffenet slightly outperforming (less than 1%) GoogLeNet is suggestive of the fact that deeper features of the latter don't contain any additional information for our small and well structured dataset. We anticipate GoogLeNet to outperform Caffenet on a larger dataset of PAD images which have higher variance in them.

The CNN model architectures we used are originally de-

signed for general object recognition, training with thousands of images for feature extraction. However, the images that are used for training have a high level of variance, both intra-class and inter-class. Our dataset, although much smaller in size, has images which are not so random and much more structured. That might be the reason behind the feature learning mechanisms of these architectures transferring well to a specific recognition task like ours. To explore this further we decided to run some additional experiments on Caffenet.

We predicted two factors to be contributing to this high performance on Caffenet - (1) the rectification and salient region cropping process of the raw PAD images as described in section 3 and (2) the rigid ordering of the reagents in the lanes. To check the influence of the rectification process on the performance, we built a dataset with unrectified wild type images for the same 26 drugs and optimal reagents. An example image can be seen in Figure 9.a. The images are directly used for training and testing, without any cropping of the salient region as well. We train Caffenet with 520 images as before and test with the remaining 260. The result can be found in Table 2. Although the accuracy in this case drops a little, it still does a better job in classifying the drugs than some of the other methods do with even rectified images. The filter maps, when pulled out from the network, suggest that the network locates the salient regions automatically even from the unrectified images (Figure 9.b).

To examine the effect of the rigid ordering of the reagents in the vertical lanes has on the classification performance, we perturb the structure of the PAD images again. We randomly re-order the lane positions in the training and testing salient crops for each drug such that the position of a reagent (lane A - lane L) in the training images is different than its position in the test images. A pair of example training and testing images can be seen in Figure 10. We train Caffenet as before, with 520 perturbed images and test it with 260 images. The accuracy goes down significantly in this case, as shown in Table 2. The result clearly indicates that the reordering of the reagents hampers Caffenet's learning process as the features captured during training are much different from the test features. Thus, the network indeed looks at blob patterns in the images and not just the color tones when extracting features.

Label correspondence:-
1 - Acetaminophen
2 - Acetylsalicylic Acid
3 - Amodiaquine
4 - Amoxicillin
5 - Ampicillin
6 - Artesunate
7 - Calcium Carbonate
8 - Corn Starch
9 - Diethylcarbamazine
10 - Ethambutol
11 - Isoniazid
12 - Rifampicin
13 - Tetracycline
14 - Azithromycin
15 - Chloramphenicol
16 - Chloroquine
17 - Ciprofloxacin
18 - DI Water
19 - Dried Wheat Starch
20 - Penicillin G
21 - Potato Starch
22 - Primaquine
23 - Quinine
24 - Streptomycine
25 - Sulfadoxine
26 - Talc

Average Prediction Confidence (out of 10) vs Actual Drug Label

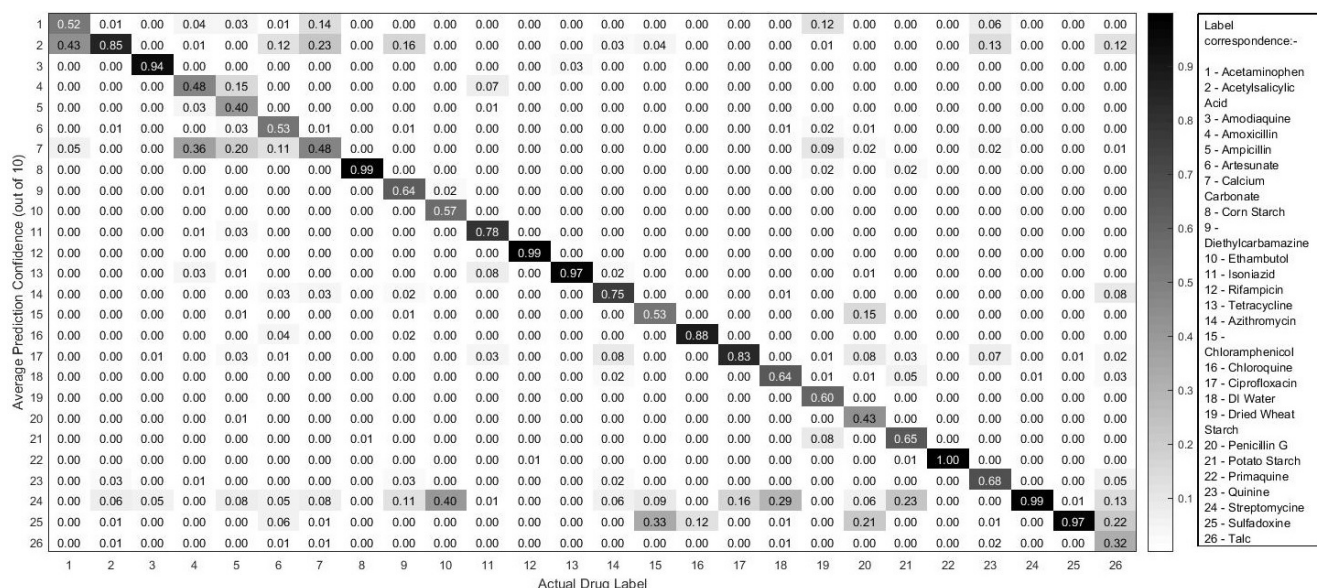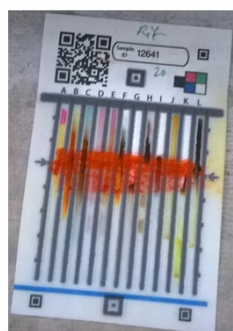| Pred\Actual	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
1	0.52	0.01	0.00	0.04	0.03	0.01	0.14	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.12	0.00	0.00	0.00	0.06	0.00	0.00
2	0.43	0.85	0.00	0.01	0.00	0.12	0.23	0.00	0.16	0.00	0.00	0.00	0.00	0.03	0.04	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.13	0.00	0.12
3	0.00	0.00	0.94	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	0.00	0.00	0.00	0.48	0.15	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	0.00	0.00	0.00	0.03	0.40	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
6	0.00	0.01	0.00	0.00	0.03	0.53	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	0.05	0.00	0.00	0.36	0.20	0.11	0.48	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.09	0.02	0.00	0.00	0.02	0.00	0.01
8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.02	0.00	0.00	0.00	0.00
9	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.64	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.57	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
11	0.00	0.00	0.00	0.01	0.03	0.00	0.00	0.00	0.00	0.00	0.78	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
13	0.00	0.00	0.00	0.03	0.01	0.00	0.00	0.00	0.00	0.00	0.08	0.00	0.97	0.02	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00
14	0.00	0.00	0.00	0.00	0.00	0.03	0.03	0.00	0.00	0.02	0.00	0.00	0.00	0.75	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08
15	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.53	0.00	0.00	0.00	0.00	0.15	0.00	0.00	0.00	0.00	0.00	0.00
16	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.02	0.00	0.00	0.04	0.00	0.00	0.00	0.88	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
17	0.00	0.00	0.01	0.00	0.03	0.01	0.00	0.00	0.00	0.08	0.00	0.00	0.00	0.08	0.00	0.00	0.83	0.01	0.08	0.03	0.00	0.07	0.00	0.01	0.01	0.02
18	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.64	0.01	0.01	0.05	0.00	0.00	0.01	0.00	0.03
19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.60	0.00	0.00	0.00	0.00	0.00	0.00	0.00
20	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.43	0.00	0.00	0.00	0.00	0.00	0.00
21	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.00	0.65	0.00	0.00	0.00	0.00	0.00
22	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	1.00	0.00	0.00	0.00	0.00
23	0.00	0.03	0.00	0.01	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.68	0.00	0.00	0.05
24	0.00	0.06	0.05	0.00	0.08	0.05	0.08	0.00	0.11	0.40	0.01	0.00	0.00	0.06	0.09	0.00	0.16	0.29	0.00	0.06	0.23	0.00	0.00	0.99	0.01	0.13
25	0.00	0.01	0.06	0.00	0.00	0.06	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.33	0.12	0.00	0.00	0.00	0.21	0.00	0.00	0.01	0.00	0.97	0.22
26	0.00	0.01	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.32

Figure 8: The average prediction confidence for classifying each drug class in Caffenet. The diagonal values represent the prediction confidence for that drug index.
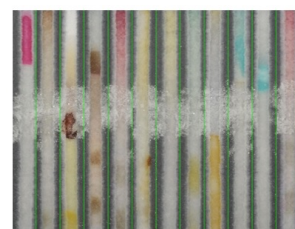
(a)

(b)

Figure 9: (a) A sample unrectified wild type PAD image for the drug *rifampicin*, (b) the filter response generated for unrectified card by the 2nd convolutional layer (*conv2*) in Caffenet.

(a)

(b)

Figure 10: (a) A perturbed training image for the drug *acetaminophen*, (b) a perturbed testing image for the same drug. It can be easily observed that the lane positions are different for the two images.

6. Conclusion

In this paper, we have proposed the novel problem of visual classification of pharmaceutical drugs and its application in counterfeit drug detection. Unlike prior research in this area, our work doesn't use the shape, color or imprint information or the pill appearance itself for this purpose. Instead we focus on the chemical composition of the drug by observing the color tones it generates reacting with different reagents on a PAD card, like a chemical fingerprint of the drug. This facilitates the detection of fake drugs, as the change in chemical composition can easily be picked up by the chemical tests. We developed a dataset of 780 PAD cards with 26 drugs and 11 optimal reagents (and a timer agent) to conduct our experiments. The raw PAD images were aligned using a rectification process and the salient region from each image was cropped out. We extracted hand-crafted features from the salient crops and trained kNN and SVM classifiers with them. We also trained two popular CNN architectures (Caffenet and GoogLeNet), originally designed for general object recognition, with these images. Both the models performed well in accurately classifying the test card images with Caffenet giving the highest average accuracy of over 94% on our dataset. On experimenting with unrectified and unstructured PAD images on Caffenet it was observed that the order of the different reagents in the vertical lanes of the cards is a vital factor in accurate classification of the drugs.

While our research has made a promising start, more

work is needed in certain areas. To improve the classification performance we require more data *i.e.* more physical cards to take the images from. Our collaborators in Africa have been sending in their test images (actual field data) recently and we are in the process of making a bigger database for classification with more drugs. The other option is to automate the card preparation process. Currently we manually bake, embed reagents, swipe drug and test the cards by hand in a wet lab, which is both time-consuming and tedious. We plan to automate this so that the cards are manufactured faster and in a more streamlined manner. The other possible extension of our work would be to design our own CNN architecture, instead of using one out of the box. Although Caffenet performs well in classifying the PAD images, still its originally designed for general object recognition. A network designed for a more specific object classification, like ours, should improve the performance. Lastly, we are in the process of implementing a message service based cell phone application through which any person in possession of a PAD card would be able to send us their test images. This will be the best way to get our hands on large quantities of images, including different brands of pharmaceutical products and falsified drugs, to improve the visual recognition system. If just one sample of a very poor quality medication is found using the system, additional targeted sampling, analysis, and regulatory action can be carried out to get the affected products off the market.

References

[1] J. P. Renschler, K. M. Walters, P. N. Newton, and R. Laxminarayan, "Estimated Under-Five Deaths Associated with Poor-Quality Antimalarials in Sub-Saharan Africa", in *American Journal of Tropical Medicine and Hygiene*, 2015, 92(6), pp. 119 - 126. 1

[2] BBC article on fake drug spotting. Available at http://www.bbc.com/news/health-32938075. 2

[3] A. Weaver, H. Reiser, T. Barstis, M. Benvenuti, D. Ghosh, M. Hunkler, B. Joy, L. Koenig, K. Raddell, and M. Lieberman, "Paper analytical devices for fast field screening of beta lactam antibiotics and anti-tuberculosis pharmaceuticals", in *Analytical Chemistry*, 2013, 85 (13), pp. 6453 - 6460. 1

[4] A. Hartl, C. Arth, and D. Schmalstieg, "Instant Medical Pill Recognition on mobile phones", in proc. of *IASTED International Conference on Computer Vision*, 2011, pp. 188 - 195. 2

[5] R.C. Chen, Y.K. Chan, Y.H. Chen, C.T. Bau, "An automatic drug image identification system based on multiple image features and dynamic weights", in *International Journal of Innovative Computing, Information and Control*, 2012, 8 (5), pp. 2995 - 3013. 2

[6] Z. Chen, and S. Kamata, "A new accurate pill recognition system using imprint information", in proc. of *SPIE International Conference on Machine Vision*, 2013, 906711. 2

[7] Y. B. Lee, U. Park, A. K. Jain, and S. W. Lee, "PILL-ID Matching and Retrieval of Drug Pill Imprint Images", in *Pattern Recognition Letters*, 33 (2011), pp. 904 - 910. 2

[8] Pill Identification Wizard from Drugs.com. Available at http://www.drugs.com/imprints.php. 2

[9] Pill Identification - WebMD: Identify Drugs by Picture, Shape, Color or Imprint - Pill Identifier with Pictures. Available at http://www.webmd.com/pill-identification. 2

[10] Pill Identifier: Identify Drugs by Picture, Shape, Color, Number. Available at http://www.healthline.com/pill-identifier. 2

[11] WHO List of Essential Medicines. Available at http://www.who.int/medicines/publications/essentialmedicines/en/. 3

[12] The OpenCV library. Available at http://opencv.org/. 3

[13] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge", in *International Journal of Computer Vision*, 2015, pp. 1 - 42. 5

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", in proc. of *Advances in Neural Information Processing Systems*, 2012. 5

[15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going Deeper with Convolutions", in *arXiv preprint arXiv:1409.4842*, 2014. 5

[16] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional Architecture for Fast Feature Embedding", in *arXiv preprint arXiv:1408.5093*, 2014. 5

[17] A. Khosla, J. Xiao, A. Torralba, and A. Oliva, "Memorability of Image Regions", in proc. of *Advances in Neural Information Processing Systems*, 2012. 5

[18] J. van de Weijer, C. Schmid, and J. Verbeek, "Learning Color Names from Real-World Images", in proc. of *IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1 - 8. 5

[19] R. Khan, J. van de Weijer, F. Khan, D. Muselet, C. Ducottet, and C. Barat, "Discriminative Color Descriptors", in proc. of *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2866 - 2873. 5

[20] C. Elkan, "Using the triangle inequality to accelerate k-means", in proc. of *International Conference on Machine Learning*, 2003. 5

[21] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification", in proc. of *IEEE Conference on Computer Vision and Pattern Recognition*, 2010. 5

[22] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories", in proc. of *IEEE Conference on Computer Vision and Pattern Recognition*, 2006. 5

[23] A. Oliva, and A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope", in *International Journal of Computer Vision*, 2001. 5

[24] M.-K. Hu, "Visual pattern recognition by moment invariants", in *IRE Transactions on Information Theory*, 1962, 8 (2), pp. 179 - 187. 2

[25] D. Lowe, "Distinctive image features from scale-invariant keypoints", in *International Journal of Computer Vision*, 2004. 2, 5

[26] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns", in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, 24 (7), pp. 971 - 987. 2

[27] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database", in proc. of *IEEE COnference on Computer Vision and Pattern Recognition*, 2009. 6