Automatic Visual Inspection of Corn Kernels Using Principal Component Analysis

Paulus Potter,¹ José M. Valiente,² Gabriela Andreu-García²

¹Eindhoven University of Technology, PO Box 513, 5600 MB, Eindhoven (The Netherlands)
²Institute of Control Systems and Industrial Computing. Universitat Politècnica de València (UPV) Camino de Vera (s/n), 46022 Valencia (Spain)
E-mail: p.potter@student.tue.nl, {jvalient,gandreu}@disca.upv.es

Abstract

This paper presents an approach that combines algorithm-based computer vision techniques and principle component analysis (PCA) to eliminate poor quality corn kernels. Experiments show that the method is promising (89% success) but extensions are recommended to further improve results.

Keywords: food inspection, computer vision, principal component analysis

1. Introduction

Many different types of corn defects exist (US Federal Grain Inspection Service, 2007). Inspections for heat-damage or damaged corn kernels are still performed by human experts. The difference between acceptable or rejectable corn kernels may be small and damage and genetic differences lead to a great unpredictability in colors and textures (Figure 1). These factors and other minor differences that may exist create a major challenge for automatic corn inspection.

FIGURE 1: Example of purple plumule (no damage) vs. blue-eye mold (damaged) and two examples of insect damage (US Federal Grain Inspection Service, 2007). A schematic diagram of an automatic corn grain classifying system is shown on the right.

Computer vision systems are being used increasingly in the food industry for quality assurance (T. Brosnan et al., 2004) and variety identification (X. Chen, et al. 2010). These systems offer the potential to automate manual grading practices thus standardizing techniques and eliminating tedious human inspection tasks.

This paper presents a method to determine if a corn kernel is acceptable or rejectable, using a PCA-based novelty detection approach. Many different novelty detection methods exist: classification-based, nearest-neighbor, clustering, statistical, as well as
spectral decomposition (M. Markou et al., 2003). Model data approaches without assumptions can also be used – such as neural networks, non-parametric statistical techniques, and spectral decomposition. Statistical approaches offer the advantage of being cheap to compute, but the amount and quality of the training is crucial (M. Markou, 2003). A. Ferrer et al. (2006) use a spectral decomposition statistical approach based on principal component analysis (PCA) to detect defects in random color textures (F. Lopez, et al. 2010).

2. Materials and methods

We propose an automatic corn grain classifying system as shown in Figure 1. The system is separated in two parts: acquisition and actuators. In this paper we focus on the acquisition part. The setup is composed of a guiding tube, a smart camera, two illumination rings, and a laser barrier. The acquired images are sent to a remote computer and the quality inspection algorithm is performed by MATLAB.

2.1 PCA method

PCA is a classical method for feature selection in pattern recognition. When it is used with images, PCA composes the original raw pixels into a number of uncorrelated variables called principal components (PCs) that describe the relevant information about the image in decreasing order. These components form an 'image model' that can be used in a novelty-detection framework. In this work we propose a method similar to the method described by A. Ferrer et al. (2006) and composed of two stages: training and testing. The training and test algorithms involve similar steps as can be seen in Figure 2.

\[
x_{ij} = r_{ij}, r_{(i-1,j-1)}, \ldots, r_{(i-1,j)}, g_{ij}, g_{(i-1,j-1)}, \ldots, g_{(i-1,j)}, b_{ij}, b_{(i-1,j-1)}, \ldots, b_{(i,j-1)}
\]

(1)
Other color spaces such as HSV or CIELAB can also be used. As result, we have an unfolded matrix $X$ of dimension $mxL$, consisting of $m$ feature vectors of good corn points with $L=27$ color-spatial components each.

The PCA is performed on the unfolded matrix $X$ after mean centering. This is done using eigenvalue decomposition of the covariance matrix $\Sigma=XX^T$, which gives a reference eigenspace $E$ formed by the eigenvectors of $\Sigma$. This orthogonal matrix $E$ that is sized $LxL$ represents our ‘model’ of good corn points.

Using the reference eigenspace, we can calculate the $T^2$ Hotelling score which represents the fit of each sample point to the model. In fact, $T^2$ is the squared Mahalanobis distance from the sample points to the center of the eigenspace. By projecting the training image onto the reference eigenspace, a score matrix $A(mxL)$, used to calculate the Hotelling $T^2$ value, is obtained as follows:

$$X = AE^T + R \quad (2) \quad A =XE \quad (3)$$

$X$ is the mean-centered color-spatial feature matrix, $E$ the eigenspace, and $R$ the residual matrix. $R$ contains noise in the data and is not taken into account in subsequent calculations (F. Lopez et al. 2010). Using the score matrix $A$, the Hotelling $T^2$ value for a pixel $i$ can be calculated as follows:

$$T_i^2 = \sum_{l=1}^{L} \frac{a_{il}^2}{s_l^2} \quad (4)$$

$a_{il}^2$ is the score value of a given pixel $i$ in the $l$th principal component, and $s_l^2$ is the variance measured in that component in the score matrix $A(mxL)$.

With $T^2$ an acceptable or defective pixel can be detected using a threshold $\tau$. To determine this threshold, a cumulative histogram of the $T^2$ values from the training stage samples is computed. A threshold $\tau$ is then determined by choosing a $T^2$ value below which a certain percent of observations fall. The rank of this percentile (typically 90% or higher) is a parameter of the algorithm.

The test stage involves a similar process as the training stage. A test sample is unfolded and projected onto the reference eigenspace according to:

$$A_{\text{new}} = X_{\text{new}}E \quad (5)$$

$X_{\text{new}}$ is the unfolded test image matrix. The $T^2$ values are again calculated and compared with the $\tau$ threshold determined earlier. Every test point falling below the threshold is considered as normal (non-defective), but those above the threshold are considered as novelties (defects). The result is a map of detected defective corn points, like those shown in figures 4 and 5. In the images on the left the points with a $T^2$ that is higher than the threshold are marked. On the right, a hot map is shown; where red and yellow indicate high $T^2$ values and blue indicates low $T^2$ values.

2.2 Grain segmentation

After acquiring a color image, corn kernels must be isolated from their background. Initially we used a thresholding method but the results were poor. Finally, the above PCA procedure was also used to segment the corn kernel from the image. The captured image is firstly projected in a reference eigenspace that corresponds to the background without the corn kernel. The novelty points then correspond to corn.
Finally, the extracted corn kernel is triple eroded with a disk structuring element to remove edges. This method for extracting backgrounds is known as eigenbackgrounds and has been previously used for detecting moving objects (M. Piccardi, 2004).

3. Discrimination of corn grain quality

The above procedure does not allow us to determine the quality of whole corn kernels as acceptable or unacceptable, and so we introduce the idea of an acceptable quality limit. Among other possibilities, the corn quality is defined as the relative percent of defective points detected in the image \( i \), which is the error rate \( \varepsilon_i \) obtained on applying the PCA model:

\[
\varepsilon_i = \frac{\text{# defective points}}{\text{# total points of corn kernel}} ; \quad \varepsilon_{\text{average}} = \frac{1}{n} \sum_{i}^{n} \varepsilon_i ; \quad (6)
\]

As we have \( n \) training images, we can obtain an averaged value \( \varepsilon_{\text{average}} \) that represents the mean error reported during training. By adding an extra margin we then define the acceptable quality limit as \( \rho = \varepsilon_{\text{average}} \ast 1.1 \).

Finally, a corn image is accepted or rejected using \( \rho_{\text{test}} \) (the error rate of the tested corn kernel) using the following simple rule:

"Accept if \( \rho_{\text{test}} \leq \rho \) and reject otherwise".

4. Experiments and results

For the training stage 20 images of high quality corn kernels were acquired. To test the algorithm we built a database of 400 images: 200 good and 200 bad corn kernels. Kernel pictures are taken from the front or the back but not from the side. All corn kernels were classified by a human expert from a corn-related food company.

A corn kernel can have both dark and light areas; either part may deviate too far from the training model and so be unjustly rejected. For this reason the experiments were carried out in three color spaces (RGB, HSV, CIELAB) and using five histogram percentile rank values: 90th, 92.5th, 95th, 97.5th and 99th.

<table>
<thead>
<tr>
<th>Color space</th>
<th>RGB</th>
<th>HSV</th>
<th>CIELAB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( t_p )</td>
<td>( f_p )</td>
<td>( t_p )</td>
</tr>
<tr>
<td>90th</td>
<td>169</td>
<td>79</td>
<td>185</td>
</tr>
<tr>
<td>95.5th</td>
<td>164</td>
<td>79</td>
<td>188</td>
</tr>
<tr>
<td>95th</td>
<td>160</td>
<td>67</td>
<td>188</td>
</tr>
<tr>
<td>97.5th</td>
<td>148</td>
<td>41</td>
<td>185</td>
</tr>
<tr>
<td>99th</td>
<td>132</td>
<td>30</td>
<td>177</td>
</tr>
</tbody>
</table>

4.1. Results

The outcomes of the experiments are shown in Table 1, where: \( t_p \) are the pixels correctly marked as rejectable; \( f_p \) the pixels not rejectable but marked as such; \( f_n \) are the rejectable but undetected pixels; and \( t_n \) the pixels correctly marked as acceptable. The results shown in Table 2 use the three statistical measures that are commonly used in binary classification: precision \( P \); recall \( R \) (or true positive rate, TPR); and accuracy F-score \( F \):

\[
P = \frac{t_p}{t_p + f_p} ; \quad R = \frac{t_p}{t_p + f_n} ; \quad F = 2 \ast \frac{P \ast R}{P + R} ; \quad (7)
\]
The results show that the highest F scores are associated with the HSV color space and a histogram percentile of 97.5th or 99th. To make the differences clearer, a receiver operating characteristic (or ROC curve) can be made that normally uses the TPR and false positive rate (FPR). The FPR is defined as:

$$FPR = \frac{f_p}{f_p + t_n} \quad (8)$$

<table>
<thead>
<tr>
<th>Percentile rank</th>
<th>RGB</th>
<th>P</th>
<th>R</th>
<th>F</th>
<th>HSV</th>
<th>P</th>
<th>R</th>
<th>F</th>
<th>CIELAB</th>
<th>P</th>
<th>R</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>90th</td>
<td>0.68</td>
<td>0.84</td>
<td>0.75</td>
<td>0.76</td>
<td>0.64</td>
<td>0.57</td>
<td>0.60</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>95th</td>
<td>0.70</td>
<td>0.80</td>
<td>0.75</td>
<td>0.79</td>
<td>0.75</td>
<td>0.58</td>
<td>0.66</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>97.5th</td>
<td>0.78</td>
<td>0.74</td>
<td>0.76</td>
<td>0.85</td>
<td>0.76</td>
<td>0.59</td>
<td>0.67</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>99th</td>
<td>0.81</td>
<td>0.66</td>
<td>0.73</td>
<td>0.89</td>
<td>0.80</td>
<td>0.57</td>
<td>0.66</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The ROC curve shows (Figure 3) the results of the three color spaces, with the solid line indicating the classification if the algorithm chooses randomly. The encircled point corresponding to HSV-99th is the point closest point to (0,1), or the point of perfect classification. Hence, HSV-99th can be considered the best in this experiment. For RGB and CIELAB this is 97.5th.

![FIGURE 3: ROC curve with the black circle indicating the optimum.](image)

![FIGURE 4: Different results obtained with RGB, HSV, and CIELAB color spaces.](image)

5. Discussion

As seen in the previous section, HSV with a percentile of 99% gives the best results, CIELAB performs the worst in detecting rejectable corn kernels and RGB gives relatively good results for corn kernel detection at lower histogram percentiles, but also produces a high false positive detection rate. Figure 4 shows an acceptable corn kernel with a histogram value of 99%, and we see that RGB and CIELAB produce marks on the germ of the corn, whereas HSV has the tendency to put marks on the transition between germ and the yellow part of the corn. The differences are not very large on the RGB and CIELAB hot map. It seems that \( p \) can be a sensitive parameter.

We expected the algorithm to classify all rejectable corn kernels as rejectable, but results show this is not the case if the corn kernel contains textures and colors that
appear to be correct in the trained model. Figure 5 shows that the transition between the lighter germ part and the cap, as well as the dark parts of the cap, is mainly marked as rejectable. It seems that PCA cannot capture these areas very well.

![Figure 5: Acceptable corn kernel marked rejectable using HSV](image)

In conclusion, it seems that the PCA model is not accurate enough. We have therefore used a retraining method to produce a more accurate model. We use one image that also contains these darker areas of a corn kernel and add the marked areas ten times to the unfolded matrix. Initially 3.36% is marked as rejectable and we retrain the model until there is no further improvement in the error rate. This appears to be at 1.8% after 262 iterations. We then tested the new PCA model on the 400 sample images using HSV-99th as this gave the best results in the other experiments. False detections decreased from 23 to 15 (tp=139, fp=15). However, the correctly classified corn kernel rate also dropped from 89% to 81%.

6. Conclusions and future work

In this work we have explored an algorithm using PCA to classify corn kernels. Experiments have been done using 400 corn kernels classified by experts. The results show that the HSV color space and a 99% histogram threshold give the best results: with 89% being correctly classified. Problems appear when corn kernels are broken (leaving the textures and colors intact).

The problem of classifying corn kernels is not easily solved and a simple PCA model is insufficient. A multi-model (two or more) solution for darker and lighter parts will probably be necessary. It is also likely that different models will be needed to handle each type of defect.

Reference list


