Evaluating Freshness of Produce Using Transfer Learning

Antony Lam and Yoshinori Kuno
Graduate School of Science and Engineering
Saitama University
255 Shimo Okubo, Sakura-ku, Saitama 338-8570, Japan
Email: {antonylam,kuno}@cv.ics.saitama-u.ac.jp

Imari Sato
Digital Content and Media Sciences Research Division
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-0003, Japan
Email: imarik@nii.ac.jp

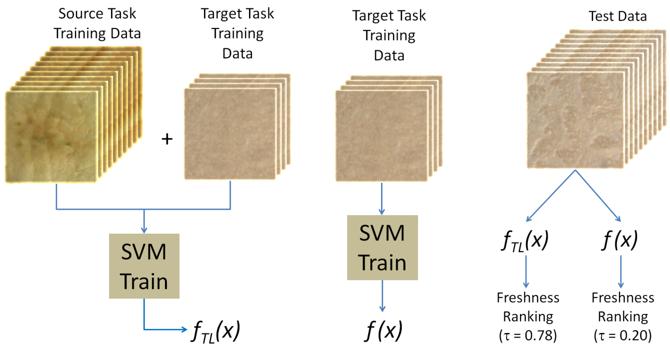
***Abstract*—Automated quality control of produce such as fruits and vegetables is of great importance to industry. In particular, the ability to evaluate the state of decay for various produce items would allow for efficient sorting of produce such that the freshest items could be more quickly shipped to consumers. Unfortunately, training an accurate classifier for determining how decayed produce is can require a large amount of data. This problem is further exacerbated by the large variety of produce available as different items would exhibit decay in different ways. In this paper, we propose an algorithm that can learn an accurate ranking classifier for sorting produce using only a small amount of data. We achieve this through our proposed transfer learning algorithm that is able to automatically select good preexisting source task training data to supplement insufficient training data in the given target task. We show how much our algorithm improves over standard training on real images of produce items captured at various stages of decay.**

I. Introduction

Automated quality control of produce is already an integral part of modern agriculture. In particular, automated sorting and grading of produce has been shown to be essential to major processing tasks in the production of fresh-market fruits [1]. In addition to being able to ship fresh goods to consumers, sorting also helps to avoid spoilage by having similarly classed items stored together [2]. (For example, rates of ripening in some fruits can be affected by proximity to other fruits.)

There are a number of existing systems for evaluating the quality of fruits and vegetables as evidenced in different survey papers [1]–[3]. Many of these proposed systems include machines that use computer vision to check the color, size, and texture of the items. While effective, these systems were designed to only work with specific target products such as apples, bananas, and pomegranate. The survey paper [3] has an extensive set of sections describing many recent systems for evaluating particular types of fruit. Given the wide variety of items on the market, the need to custom tailor systems for each type of item can be expensive. It would be good if one were able to build a single system that could quickly learn how to sort any produce by only observing a few samples of fresh and decayed items. For example, it would be good if we could simply show a sorting machine four examples of something exotic like physalis fruits at different stages of ripeness and have it be able to sort all other physalis fruits.

An obvious way to build such a general system would be to base it on machine learning. One could take the samples of produce at different stages of decay and train a classifier such as a Ranking SVM [4], [5] to build a visual freshness ranker. The Ranking SVM could then be used to sort new (unlabelled) samples of the target produce. Unfortunately, good generalization performance from just a few examples (e.g. five) is not always possible.



(a) Transfer Learning (b) Standard Learning (c) Freshness Ranking

Fig. 1. Transfer learning for ranking cut potatoes by freshness. (a) Training data on how cut apples decay is used to supplement the small training dataset on how potatoes decay. (b) Training only using the few potato examples. (c) Compared freshness ranking performance using transfer learning $f_{TL}(x)$ and without transfer learning $f(x)$. The Kendall tau rank correlation coefficient indicates transfer learning improved results. (1 is ideal.)

Transfer learning [6] aims to overcome the inability to generalize from insufficient training data for a target task by supplementing the missing data using one or more related source tasks. For example, if our target task were to learn how to sort potatoes from only a few examples, we might be able to supplement the small amount of training data with a large amount of labelled data about apple decay. This is because there may be shared appearance characteristics between how potatoes and apples decay. Fig. 1 illustrates this process.

However, another problem is that we may have a large number of source task data and it is difficult to know which source tasks would help with learning the target task. For example, we may have data on apples, oranges, bananas, beef steak, and so on. If we choose irrelevant source task data for transfer learning, we may end up with a worse classifier than one trained without transfer learning. How can we choose a source task such that performance on novel examples is improved? This problem is called the source task selection problem and it is an area of active research. In our paper, our main contribution is that we propose a new source task selection algorithm. We then show that once the source task is selected, a relatively straightforward form of transfer learning via weighting of training data can be used to improve generalization performance for evaluating freshness of produce using real images.

II. Related Work

There have been a number of computer vision based methods for evaluating produce [1]–[3]. For example, specific systems for evaluating apples [7], grapes [8], bananas [9], watermelons [10], and oranges [11] have been proposed. Development of these systems involved careful selection of mostly color and texture features and the training of different machine learning classification algorithms for the one type of target produce to be evaluated. For example, [9] trained a neural network on RGB histograms and evaluated bananas. While effective, these systems only work for the specific type of produce they were trained to evaluate.

Since there are a very large variety of produce items on the market, the training of individual classifiers for specific types of items is expensive. This is because training an effective classifier for each specific type of item (e.g. apples, oranges, physalis fruits) would typically require a large amount of training data. If we could train a classifier with much less data, this would allow for a given evaluation system to be quickly trained for many more types of produce items. A solution to this problem is to use transfer learning [6]. Transfer learning overcomes the limitations of learning a target task from a small amount of training data by supplementing the missing information using preexisting information from related source tasks. For example, if one had very little training data for evaluating lemons but had a large amount of data on oranges, the orange data may be used to help learn a better lemon evaluation system.

However, given a large number of source tasks (e.g. evaluators for oranges, bananas, potatoes, beef steaks), there is the question of how we can automatically decide which source task would be most helpful for transfer learning. As described in the introduction, choosing the wrong source task can give worse performance. In this paper, we propose a fully automatic source task selection method. We show in experiments on real images that we can improve produce freshness ranking performance using only a small amount of training data.

III. Learning Formulation

For the rest of the paper, we will represent produce items as fixed length vectors $\vec{x}$. These could be constructed directly from the raw images of the items or via some form of feature extraction such as a histogram of pixel values. The details of how we extract features can be found in Sec. IV.

A. Ranking SVM

For learning to rank, we use the Ranking SVM [4], [5]. The Ranking SVM formulation, is similar to the standard SVM formulation for binary classification. The difference is that instead of learning binary labels associated with training data, binary preferences between pairs of data $\vec{x}_i$ and $\vec{x}_j$ are learned. The optimization can be written as

$$\begin{gathered} \min V(\vec{w},\vec{\xi}) = \frac{1}{2}\vec{w}\cdot\vec{w} + C\sum_{i,j}\xi_{i,j} \\ s.t.\ \forall_{i,j}\in r,\ \vec{w}\phi(\vec{x}_i) \geq \vec{w}\phi(\vec{x}_j) + 1 - \xi_{i,j},\ \xi_{i,j}\geq 0 \end{gathered} \tag{1}$$

where $\vec{w}$ is a weight vector, C is the trade-off parameter between margin size and training error, $\vec{\xi}$ is the vector of all slack variables, ϕ is the kernel function, and r is the set of ranking preferences between i and j such that $\vec{x}_i$ is ranked higher or equal to $\vec{x}_j$. We then take $f(\vec{x}) = w\phi(\vec{x})$ as our Ranking SVM where given input from a single produce item $\vec{x}$ (e.g. image data from an apple), we get a scalar value as output. In our formulation, fresher produce items have larger associated scalar values so a higher output value ranks the item higher on the list.

B. Transfer Learning from a Given Source Task

For the moment, let us assume that the source task training dataset S has already been given. Then given the target task training dataset T (which is a very small dataset), we perform transfer learning by training the Ranking SVM using both datasets S and T together. However, rather than naively combining the datasets together and simply using Eq. (1), we also reformulate the optimization so training errors on data from S effectively have low weight while training errors on data from T have high weight. The rationale for weighting the training errors is that we do not want data from the much larger set S to possibly cause the SVM to ignore the data from the much smaller set T. Formally, we only make a slight modification to Eq. (1):

$$\begin{gathered} \min V(\vec{w},\vec{\xi}) = \frac{1}{2}\vec{w}\cdot\vec{w} + C_T\sum_{i,j}\xi_{i,j} + C_S\sum_{k,l}\xi_{k,l} \\ s.t.\ \forall_{i,j}\in r_T,\ \vec{w}\phi(\vec{x}_i) \geq \vec{w}\phi(\vec{x}_j) + 1 - \xi_{i,j},\ \xi_{i,j}\geq 0 \\ \forall_{k,l}\in r_S,\ \vec{w}\phi(\vec{x}_k) \geq \vec{w}\phi(\vec{x}_l) + 1 - \xi_{k,l},\ \xi_{k,l}\geq 0 \end{gathered} \tag{2}$$

where C_T and C_S are the trade-off parameters between the margin size and training error for datasets T and S respectively, and r_T and r_S are the ranking preferences for the two tasks. Subscripts i, j and k, l are indices to specific instances (produce items such as apples and potatoes) in their respective datasets T and S respectively. We set C_T to be very large and C_S to be small in order to impose a much higher penalty on training errors on dataset T. To differentiate the Ranking SVM trained from Eq. (2), we denote it as $f_{TL}(\vec{x})$ as opposed to the standard Ranking SVM $f(\vec{x})$ from Sec. III-A.

C. Source Task Selection

The automated selection of a good source task is also important for transfer learning to be effective. For example, if we wanted to better rank potatoes on freshness, using a source task for ranking beef steaks could even lower performance. We propose a new approach for selecting a good source task that was inspired by [12] where the basic idea was to see which source task SVM (trained without the target task) could classify the target task training dataset best. This would provide an indication of which source task is most related to the target task. The authors in [12] developed their method using a binary classification formulation but we extend their idea to a ranking formulation in this paper.

In order to evaluate which rankings are best, we use the Kandall tau rank correlation coefficient [13]. The rank correlation basically looks at two rankings, which in our case is the ground truth ranking and the SVM's ranking, and compares them in terms of how many agreements versus disagreements there are between the two rankings. Formally, it is defined as

$$\tau = \frac{n_c - n_d}{\frac{1}{2}n(n-1)} \tag{3}$$

where n_c is the number of concordant pairs between the two rankings (agreements), n_d is the number of discordant pairs (disagreements), and n is the number of produce items considered. If the SVM's ranking is exactly the same as the ground truth, $\tau = 1$, if one ranking is the complete reverse of the other, $\tau = -1$. When $\tau = 0$, it means the rankings are independent of each other.

Our source task selection algorithm is simple. Given a set of source task datasets $\{S_1, S_2, ..., S_n\}$, we train Ranking SVMs $\{f_1, f_2, ..., f_n\}$. We then rank the instances $x_i \in T$ from the target task using each SVM f_k for $k = 1, 2, ..., n$. The SVM f_k that gave the best ranking according to Eq. (3), would then indicate that we should choose source task dataset S_k for transfer learning. Note that in the case of ties, we currently choose only one source task. In the future, we will investigate the best way to deal with ties.

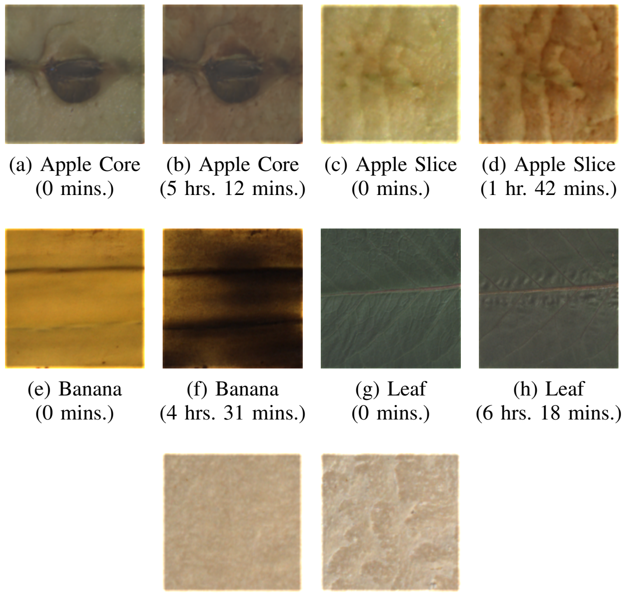
IV. Experiments

A. Features for Learning

For learning to be effective, good features need to be extracted from the raw RGB images of produce items. The many systems cited in Sec. II used color as a strong cue for differentiating between different produce items. We will also use color information.[1] We chose to construct pixel value histograms from RGB images. We then tested standard Ranking SVM training to learn rankings for freshness on five items (apple core, apple slice, banana, leaf, and potato) from the STAF: Database of Time-Varying Surface Appearance [14]. (See Fig. 2 for examples of these items.) The items in this dataset were captured under white light at time intervals ranging from 3 minutes to about 12 minutes over an average span of about 5 hours. We ultimately found that pixel value histograms of 20 bins from the red channel alone and a linear kernel for the SVM provided the best performance for all classes.

We also note that for transfer learning, we normalized the pixel values in the training images. This is because transfer learning would be ineffective if the source task had image pixel values that were say, scaled to be much larger than the values in the target task. We normalized by first taking the all training images for a given class (e.g. apple core) and computing the average pixel value for that class. All pixel values in the training images for the class were then divided by the average value. This process was performed independently for each class. For the C parameters in Eq. 2, we set $C_T = 10^9$ and $C_S = 10^{-2}$ for all experiments. We set such a large value for C_T so that we would basically perform a hard fit of the target training data.

[1]We note that features such as texture and shape were also used by past work. In our paper, we only focus on color as it is has been used most broadly for many types of produce. The emphasis of our current work is to demonstrate the benefits of our transfer learning algorithm. Future work will consider the use of multiple types of features, illuminants, multiband color images.



(a) Apple Core (0 mins.) (b) Apple Core (5 hrs. 12 mins.) (c) Apple Slice (0 mins.) (d) Apple Slice (1 hr. 42 mins.)

(e) Banana (0 mins.) (f) Banana (4 hrs. 31 mins.) (g) Leaf (0 mins.) (h) Leaf (6 hrs. 18 mins.)

(i) Potato (0 mins.) (j) Potato (5 hrs.)

Fig. 2. Sample images from the STAF: Database of Time-Varying Surface Appearance

Target Task	No TL	TL	Source Task Selected
Apple Core	0.349	0.640	Apple Slice
Apple Slice	0.963	0.968	Apple Core
Banana	0.365	0.952	Leaf
Leaf	0.293	0.280	Banana
Potato	0.200	0.776	Apple Slice
Mean	**0.434**	**0.723**	-
Std	**0.303**	**0.282**	-

Fig. 3. No transfer learning (No TL) vs. transfer learning (TL). Results show the Kendall tau rank correlation is improved with transfer learning. The automatically selected source tasks also appear to be related to the target task. For example, apple core and apple slice serve as source tasks for each other.

B. Transfer Learning

We tested transfer learning by setting each class of items as the target task and the rest of the classes as source tasks. In each target task, we only used the first five images captured in time ($t = 1, 2, 3, 4, 5$). This meant that on average, we would only have 36 minutes of decaying appearances for training but needed to accurately rank decayed samples hours in the future. For the source tasks, we used all available labelled data for transfer learning. After automatically selecting the best source task for transfer learning, the resultant Ranking SVM was used to rank an average of 29 test images from the target class. These test images were all captured later than the first five training images. Some of the test images were captured hours after the training data. See Fig. 2 for fresh and decayed examples.

Fig. 3 shows a comparison of ranking performance on the test data in the no transfer learning (No TL) and transfer learning (TL) cases. We see that the Kendall tau rank correlations are indeed improved with the use of our proposed transfer learning approach. The mean and standard deviation of the Kandall tau rank correlations also shows that without transfer learning, average performance was significantly lower and also had a slightly higher standard deviation.

Another interesting finding is that the automatically selected source tasks seem to make intuitive sense. The apple core and apple slice classes were selected as source tasks for each other even though the source tasks selections were done independently. When the target task was set to potato, our algorithm chose the apple slice source task. On visual inspection, the apple slice images (Fig. 2c-2d) do seem to be the most similar match to the potato images (Fig. 2i-2j) in terms of appearance at different stages of decay. Also note that a late stage decay image like Fig. 2j was never included in any of the training data.

The banana and leaf classes were the most different form the apple and potato images and they ended up serving as source tasks for each other. Transfer learning basically had the same performance as standard learning in the leaf ranking task. Why then did the banana ranking task benefit from the leaf source task? We speculate this is because the two classes share common characteristics but bananas have in addition, many more characteristics that are irrelevant to ranking leaves. In any case, we confirmed that the banana source task was the best possible choice from among all possible source tasks in our experiments. In a real setup, we would expect a much larger set of source tasks to choose from, which would include source tasks more closely related to the leaf ranking task.

V. Conclusion and Future Work

We presented a fully automated algorithm for learning accurate rankings for the freshness of produce from only a few examples (e.g. five). One of the key contributions of our work is the source task selection algorithm, which is able to automatically choose good source training data to supplement excessively small target training sets. A major benefit of our algorithm is that we only need to capture images of the target produce item at stages of decay over a short period of time (e.g. 35 minutes). Afterwards, our transfer learned Ranking SVM can be used to accurately evaluate the freshness of product samples even after hours of decay.

One weakness of our method is that the source task selection algorithm needs to evaluate source and target task relatedness based on how well the target training data can be ranked by a given source task SVM. This means that for stable good performance on source task selection, we need at least 4-5 target training examples. If we had too few examples, an arbitrary candidate source task SVM could achieve a good ranking on the target data completely by chance. Consider the case where we only had two examples of the target task training data. Since there would only be two possible rankings, any arbitrary SVM would have a 1/2 chance of achieving a perfect ranking! In the case of three examples, there would be six possible rankings which still means an arbitrary SVM has a 1/6 chance of getting a perfect ranking. Furthermore, there are also two other possible rankings that would tie as "near perfect" rankings. This means that there would be a total of three rankings that would either be perfect or "near perfect". As a result, an arbitrary SVM still has a 1/2 chance of achieving a very good ranking when there are only three examples. With five examples, the chance of "guessing" a perfect ranking is only 1/120. We found empirically that five target task training examples worked well to overcome chance good rankings such that reliable source tasks could be selected.

In the future, we hope to find a way to select source tasks with less data. We could consider more information than just how well a given source task SVM ranks the target training data. As mentioned earlier in the paper, another aspect of future research is to combine multiple types of features such as texture and shape. We will also investigate effective ways to combine multiple source tasks for transfer learning.

References

[1] S. M. Iqbal and A. Gopal, "Automated quality inspection of citrus fruits–a review," in *Proc. of the 2014 ICAM International Conference on Advanced and Agile Manufacturing*, 2014, pp. 1–17.

[2] D. H. Londhe, S. M. Nalawade, G. S. Pawar, V. T. Atkari, and S. V. Wandkar, "Grader: A review of different methods of grading for fruits and vegetables," *Agricultural Engineering International: CIGR Journal*, vol. 15, no. 3, pp. 217–230, 2013.

[3] J. A. Kodagali and S. Balaji, "Computer vision and image analysis based techniques for automatic characterization of fruits – a review," *International Journal of Computer Applications*, vol. 50, no. 6, pp. 6–12, July 2012.

[4] R. Herbrich, T. Graepel, and K. Obermayer, *Large Margin Rank Boundaries for Ordinal Regression*. MIT Press, January 2000, ch. 7, pp. 115–132.

[5] T. Joachims, "Optimizing search engines using clickthrough data," in *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '02. New York, NY, USA: ACM, 2002, pp. 133–142.

[6] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, October 2010.

[7] D. Unay, B. Gosselin, O. Kleynen, V. Leemans, M.-F. Destain, and O. Debeir, "Automatic grading of bi-colored apples by multispectral machine vision," *Computers and Electronics in Agriculture*, vol. 75, no. 1, pp. 204–212, 2011.

[8] "Classification of grapefruit peel diseases using color texture feature analysis," *International Journal of Agricultural and Biological Engineering*, vol. 2, no. 3, pp. 41–50, 2009.

[9] Y. Wang, Y. Cui, S. Chen, P. Zhang, H. Huang, and G. Q. Huang, "Study on fruit quality measurement and evaluation based on color identification," in *Proc. SPIE*, vol. 7513, 2009, pp. 75 130F–75 130F–6.

[10] "Classification and analysis of fruit shapes in long type watermelon using image processing," *International Journal of Agriculture & Biology*, vol. 9, pp. 68–70.

[11] "3D color co-occurrence texture features as tool to evaluate quality of fruits," *Journal of Scientific and Industrial Research*, vol. 70, pp. 912–917.

[12] A. Lam, A. K. Roy-Chowdhury, and C. R. Shelton, "Interactive event search through transfer learning," in *Tenth Asian Conference on Computer Vision*, 2010.

[13] M. G. Kendall, "A new measure of rank correlation," *Biometrika*, vol. 30, no. 1/2, pp. 81–93, 1938. [Online]. Available: http://www.jstor.org/stable/2332226

[14] J. Gu, C. Tu, R. Ramamoorthi, P. Belhumeur, W. Matusik, and S. K. Nayar, "Time-varying Surface Appearance: Acquisition, Modeling, and Rendering," *ACM Trans. on Graphics (also Proc. of ACM SIGGRAPH)*, Jul 2006.