Toward Dietary Assessment via Mobile Phone Video Cameras

Nicholas Chen, MS1, Yun Young Lee, BS1, Maurice Rabb, MS1, Bruce Schatz, PhD2
1Department of Computer Science, University of Illinois, Urbana, IL 61801 USA;
2Department of Medical Information Science, University of Illinois, Urbana, IL 61801 USA

Abstract

Reliable dietary assessment is a challenging yet essential task for determining general health. Existing efforts are manual, require considerable effort, and are prone to underestimation and misrepresentation of food intake. We propose leveraging mobile phones to make this process faster, easier and automatic. Using mobile phones with built-in video cameras, individuals capture short videos of their meals; our software then automatically analyzes the videos to recognize dishes and estimate calories. Preliminary experiments on 20 typical dishes from a local cafeteria show promising results. Our approach complements existing dietary assessment methods to help individuals better manage their diet to prevent obesity and other diet-related diseases.

Introduction

A dietary assessment is a comprehensive evaluation of a person’s food intake. It is a continuous process that measures an individual’s food and nutrient consumption history. An accurate dietary assessment provides valuable insight to an individual’s potential health problems such as malnourishment and – more common in this modern age – obesity. Complete dietary data are essential for individuals to construct personalized diet regimens to improve their eating habits for the prevention of such health issues.

Various techniques exist to aid in dietary assessment. Photo diaries are popular among individuals trying to lose weight. Individuals photograph their meals and make notes about each dish before eating. Unfortunately, a photo diary only shows what was eaten but not its nutritional value.

Calorie counting software applications are also popular. Individuals look up particular dishes in the software database to get an estimate of the nutritional contents. Though an extensive database provides greater accuracy, navigating through it creates a mental burden on the user. Did we eat fresh tomatoes or canned tomatoes? Such a meticulous approach can quickly become tedious and demotivates all but the most determined users since such choices make little caloric differences.

Performing dietary assessment using digital photos of food is becoming popular. The ubiquity of mobile phones with cameras makes photography easy and accessible. As of December 2009, there were more than 285 million mobile phones in the US alone[1]. Leveraging this, in Japan, Metaboinfo’s Virtual Wife [2] has a team of nutritionists manually analyzing mobile phone photos of dishes to provide instant calorie estimation for its users. Existing research[3,4] attempts to use automated computer vision techniques to recognize food from their photos to assist in calorie estimation. Although promising, the performance has been limited, performing optimistically at 25 - 58% accuracy. Variations caused by many factors (e.g. distance and lighting conditions) make single photos of meals poor candidates for reliable image processing. Our initial attempts at food recognition confirms this limitation, performing at less than 25% accuracy with photos taken from a bird’s-eye view.

A natural next step is to use mobile phone video cameras to acquire better images for automatic image processing. Videos provide a multi-perspective view of the food, enabling us to more reliably determine what is on the plate. Shooting video of a dish is easier and no more time-consuming than shooting a single photo because the user is relieved from needing to compose the "perfect" shot. The user simply shoots a panoramic video of the dish, and our software then selects a number of candidate frames from the video.

Furthermore, it is more robust against many environmental factors that can negatively affect the quality of photos. Switching to a video-based approach improved our accuracy up to 95%. The growing ubiquity of mobile phones with high-quality video cameras makes our approach easily deployable.

Our goal is a multistage approach toward a technology-driven, accurate and reliable dietary assessment. Such an approach will complement techniques like calorie counting software by pre-filtering irrelevant items that the user has to look up. The first stage of such an approach would be to reliably use videos to identify foods that are consumed. Future stages would derive a series of image features that are salient indicators of the nutritional characteristics of a meal: a meal with much “image texture” (spatial variation in pixel intensities) may indicate fiber; a meal that “glistens” may indicate high fat content and, thus, higher calories than a leaner meal that “glistens” less. Such features could enable direct caloric estimation.
In this paper, we demonstrate the feasibility of the first stage: identifying dishes from videos to assist in calorie estimation.

Methodology

Our approach is a pattern-matching technique. First we build a database of training images of dishes annotated with their calories. When presented with an unknown image, our system finds the best-matching images from its training set. The annotated calories from those best-matching images are then used as estimates of the calories of that image.

 Capturing Videos

We captured videos of 20 different dishes at Bevier Café, a campus cafeteria managed by the Department of Food Science and Human Nutrition at the University of Illinois. Bevier provided an ideal environment for our work. The dishes are comparable to many home-cooked meals and those served at family restaurants. Existing work[3] uses computer vision techniques on fast food meals but, to our knowledge, our attempt is the first at analyzing typical restaurant dishes which tend to have more variety than fast food.

We were given access to all of Bevier's recipes, which enabled us to calculate, by ingredient, the nutritional value of each dish. The video of the dishes was captured at 640 x 480 pixels; a typical resolution available on most mobile phones video cameras such as the Apple iPhone and Google Nexus.

In our evaluations, the dishes were placed on a horizontal turntable with a black tablecloth. The video camera was mounted on a tripod and was slanted at an angle to capture the entire dish. In practice, we envision that a user would rotate the plate manually while sitting at the table.

Food items look very different from different angles. For example, a topdown view of a panini fails to reveal the contents sandwiched in between. On the other hand, a 360 degree off-axis view captures a more representative view of the dish. We rotated the turntable manually and captured a 360 degree view of the dish. Each video is about 20 seconds long. Figure 1 shows a sample.

Figure 1. Multiple video frames of a panini dish.

Extracting Information From Videos

Once we have a video of a dish, we extract video frames from it. A video frame is a particular snapshot of that dish in time. Video frames were extracted at regular intervals to represent the dish from different angular viewpoints. Multiple still-shot photos may replace these video frames, but the additional photographic information obtainable from a video clip makes it easier to automatically determine the region-of-interest (ROI) of the dish.

Our ROI is an elliptical subsection of an image that includes as much of the food as possible but excludes much of the background and plate. Only the ROI is considered for image processing. We extract two kinds of information from the region: image features and color histogram.

Figure 2. Elliptical ROI (highlighted) and the extracted features.

We evaluated three separate computer vision algorithm for extracting image features: MSER[5], SURF[6] and STAR[7]. MSER is an algorithm for blob detection in images. Blobs are points and/or regions in the image that are either brighter or darker than the surrounding. SURF and STAR are algorithms for detecting interesting keypoints in images. Interesting keypoints are distinctive locations in the image such as corners, blobs and T-junctions. The red circles in Figure 2 show an example of image features that the SURF algorithm automatically detects for the panini dish. According to their respective authors, MSER, SURF and STAR are robust algorithms; the image features detected are scale-invariant, rotation-invariant and partially invariant to changes in illumination and geometric distortion. The robustness of these algorithms is essential for our technique: because they are rotation-invariant, the algorithms locate the same image features on a piece of food even if it has been rotated on the plate; because they are scale-invariant, the algorithms locate the same image features even if the video is zoomed in or out.

Our chosen image feature detectors only work on the monochrome channel of a video frame. Foods, however, are naturally rich in colors and this information is crucial for proper recognition. To take advantage of colors, we encode the different colors
within the ROI using a color histogram employing the HSV color model. The HSV color model is more perceptually relevant to humans than the default RGB color model, used in most electronic devices.

**Building a Vocabulary from Image Features**

We use a natural language processing model known as bag-of-words to automatically locate relevant image features. First, our system aggregates all the image feature from our collection of video frames. Then it performs k-means clustering on that data to extract 10,000 relevant features for our set of video frames; these relevant features are the cluster centers from the k-means algorithm. These 10,000 features become words in our vocabulary. Though conceptually equivalent to typical words in a language such as English, the words here are represented by 128-dimensional vectors. All video frames in our collection are then described in terms of these words. Like natural languages, the more words we have, the more descriptive we can be about a particular video frame – at the expense of more computational resources.

Existing work in computer vision shows that this bag-of-words technique is scalable up to a million images easily[8]. Thus, it is possible to build up a database of food items for different restaurants and to provide each mobile phone with such a database.

Once the system has identified the 10,000-word vocabulary, it uses Fast Library for Approximate Nearest Neighbors (FLANN)[9] to “fit” the features in each video frame into our vocabulary.

After this step, each video frame is now encoded in a common vocabulary i.e. a 10,000 bag-of-words vector. Figure 3 illustrates this encoding.

**Quantifying Similarities Between Different Video Frames**

Ultimately the goal of our technique is to be able to take a video frame of an unknown dish and determine which dish in our database matches it best. More generally, given video frames $frame_1$ and $frame_2$, we want to determine their similarities. Recall that we have two kinds of information: image features and color histograms.

**Scoring Image Features**

Describing each video frame in a 10,000 bag-of-words vector allows us to use the term frequency-inverse document frequency (tf-idf)[10] scoring technique from natural language processing.

Term frequency counts the number of times a term (word) appears in a video frame i.e. the number in the box in Figure 3 and is calculated as $\log_{10}(1 + \text{word frequency})$. Inverse document frequency counts the number of documents (video frames) that contain that particular word and is calculated as $\log_{10}(\text{Total video frames} / \text{Video frames with word})$. The tf-idf score for a word is the product of its tf score and idf score using the formulas described.

The similarity between two frames is determined by the dot product of their unit bag-of-words vectors after tf-idf scoring.

**Scoring Color Histogram**

We normalize each video frame’s histogram using $||L1 - \text{norm}||$ to account for the different sizes of their elliptical ROI. Then the correlation coefficient between the two color histograms is calculated.

**Weighted Score of Both Image Features and Color Histogram**

$$\text{Similarity} (frame_1, frame_2) = 80\% \times \text{ImageFeatures score} (frame_1, frame_2) + 20\% \times \text{Histogram score} (frame_1, frame_2)$$

We place greater emphasis on the image features as they are more robust and less affected by environmental variations. The 80%/20% heuristic was found through performance tuning on an earlier testing set (not the one in the evaluation section).

**Experimental Results**

We evaluated our technique on 20 different dishes – one salad, ten entrées, five side dishes and four desserts – covering the gamut of typical foods at a restaurant. The dishes and their calories are shown in Figure 4. Calories were calculated from the recipes using a commercial food-ingredient reference[11],
Our evaluation seeks to answer two research questions:

1) How well does our technique perform in recognizing dishes that we train it on?

We train our system on four different video frames of a dish e.g. caesar salad and tested it on five video frames of that type of dish e.g. another caesar salad. These five video frames represent a single dish taken from multiple angles. Our training set is available from [http://www.cs.illinois.edu/HealthInstrumentation/Calorie+Guru](http://www.cs.illinois.edu/HealthInstrumentation/Calorie+Guru). We use the Similarity(frame1, frame2) function defined previously. A video frame is considered correctly identified if the similarity function returns one of the four training images that corresponds to that dish as the top result. Otherwise, it is considered wrongly identified.

Because we have five video frames of a dish, we can use a voting scheme. When our system correctly identifies three out of five of the video images, it votes that the dish must indeed be that of the majority votes. This is a reasonable and effective method since it is not uncommon for a few video frames to be inconclusive while the other video frames all agree on the same dish.

Figure 4 shows our results for the three computer vision algorithms (MSER, SURF and STAR) that we evaluated. Our voting scheme comes into play for dishes such as chicken on rice, pizza, and portabelllo burger. For the chicken on rice dish, one video frame was wrongly identified in the MSER and SURF algorithms but the other four video frames agree on the same dish. Therefore, our system picks the dish that the majority agrees on. Overall, our accuracy is promising. The three algorithms performed comparably: MSER(19/20 = 95%); SURF (18/20=90%) and STAR(18/20=90%). All three algorithms wrongly identified regular fries. They confused regular fries with steak fries since those two dishes are almost identical. On the other hand, the algorithms correctly identified steak fries. This is because steak fries tend to be have more texture and our system extracted more image features that could be used to match for similarity. Even though the system confused these two dishes, the caloric contents of both are very similar. This is acceptable since our ultimate goal is to estimate the caloric content of meals as opposed to recognizing a specific dish.

2) Is our system capable of predicting a suitable match for dishes that we did not train it on?

Our system was trained only on the 20 dishes shown in Figure 4. We had two other salad dishes that the system was not trained on. We tested those two dishes on our system and it matched those two dishes as being most similar to the caesar salad in our training set. We also tested our system on a chipotle chicken on ciabatta dish. Our system thought it was most similar to the chicken on rice dish. While this match was not exact, the caloric values of both dishes are similar i.e. around 800 calories. Figure 5 shows the dishes.

<table>
<thead>
<tr>
<th>Dish</th>
<th>MSER</th>
<th>SURF</th>
<th>STAR</th>
<th>MSER</th>
<th>SURF</th>
<th>STAR</th>
<th>MSER</th>
<th>SURF</th>
<th>STAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caesar Salad (513)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>BLT Panini (360)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Calzone (403)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Chicken on Rice (858)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Pizza (463)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Portabello Burger (671)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Quiche (556)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Stew (501)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Tortellini (367)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Turkey (485)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Corn (56)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Green Beans (60)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Fried Okra (128)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Fried Olva (128)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Fried Olva (128)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Fried Olva (128)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
<tr>
<td>Fried Olva (128)</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
<td>⬤⬤⬤⬤</td>
</tr>
</tbody>
</table>

Figure 4. Results of identifying 20 dishes using MSER, SURF and STAR algorithms for feature detection.
exists a canonical set of bag-of-words that can be used to describe major dishes still remains.

Figure 5. Matching unknown dishes to dishes in the training set.

Conclusions and Future Work

We described a novel technique using mobile phone video cameras to correctly identify different foods for calorie estimation. We evaluated our technique on a variety of foods that are representative of typical meals. Using the voting scheme technique, our system is able to correctly identify many different dishes. Even on the dishes that it fails to identify, it is able to match it to a relevant dish i.e. regular fries to steak fries which has similar caloric content. Moreover, our evaluation suggests that given a large enough training set and a richer vocabulary, we would be able to match different kinds of food and make reasonable estimations for the calories of dishes that our system has not been trained on.

The methodology we presented is a first step toward the bigger goal of reliable and accurate dietary assessment. Work remains to more thoroughly evaluate this approach in the presence of illumination inconsistencies and variations in the videos captured by users. Nonetheless, our current result serves as a baseline to compare future approaches.

Our goal is to develop a system to aid dietary assessment for general health. Computer vision techniques are just one small part of that system. We intend to supplement our system with location-awareness and speech recognition techniques. Location-awareness, based on GPS information, identifies which restaurant the user is at, pruning the choice of dishes that our system has to recognize based on the restaurant’s current menu. Speech recognition will use any additional cues that the users may provide to refine caloric estimates.

The ubiquity of mobile phones and the scalability of automated techniques allow our approach to be deployable to the general population to aid dietary assessment: a user can estimate her calories consumed through her mobile phone and relate it to her calories burned from her daily activities.

Acknowledgements

Funding for this project was provided from the CIMIT Prize for Primary Healthcare. Our thanks to Richard Berlin, MD for his patient support from our project’s genesis in his (and Schatz’s) Healthcare Infrastructure course [13]. Special thanks to Jean-Louis Ledent and Jill North Craft from Bevier Café for their invaluable input, and for making their staff and the Food Service Laboratory available to us. Additional thanks to Serena Schatz, Brett Daniel, Lucas Cook and Audrey Petty.

References