# *MT-Diet*: Automated Smartphone based Diet Assessment with Infrared Images

Junghyo Lee, Ayan Banerjee, and Sandeep K. S. Gupta
IMPACT Lab, CIDSE, Arizona State University, Tempe, Az
Email: {jlee375,abanerj3,sandeep.gupta}@asu.edu

***Abstract*—In this paper, we propose *MT-Diet*, a smartphone-based automated diet monitoring system that interfaces a thermal camera with a smartphone and identifies types of food consumed at the click of a button. The system uses thermal maps of a food plate to increase accuracy of segmentation and extraction of food parts, and combines thermal and visual images to improve accuracy in the detection of cooked food. Test results on 80 different types of cooked food show that *MT-Diet* can isolate food parts with an accuracy of 97.5% and determine the type of food with an accuracy of 88.93%, which is a significant improvement (nearly 25%) over the state-of-the-art.**

## I. Introduction

Increased usage of smartphones embedded with high resolution cameras and interfaced with cost-effective wearables [1] provides us an opportunity for non-invasive automated dietary monitoring. Such a system can be ideal to curb nutrition-related epidemics such as obesity or diabetes [2].

In this paper, we propose *MT-Diet*, an automated diet monitoring smartphone application that combines infrared and color images to recognize food types on a plate of cooked food and provides feedback on relative amounts of carbohydrate, fat, fiber, and cholesterol (Fig. 1). Using *MT-Diet*, a user can take images in both infrared and visual spectrum, which are then used to identify types of food on a plate. *MT-Diet* has six essential characteristics that makes it more usable than existing diet monitoring apps: i) automated food segmentation, without any input from the user, ii) automated food identification, iii) automated food quantity estimation, using gesture recognition, iv) privacy preservation, i.e., does not depend on crowdsourcing food intake information, v) personalization, i.e., can be configured to eating habits of individuals, and vi) capacity to provide real time feedback on caloric intake. In this paper, we will evaluate two core components of *MT-Diet*: food segmentation accuracy from thermal and color images of a food plate and food identification accuracy on cooked food.

*MT-Diet* is a significant improvement from the current state-of-the-art in smartphone based diet monitoring. The existing mobile diet monitoring apps [3] can be classified into three groups using the six essential characteristics. The most basic group of apps (smartphone based journal) provides a graphical interface for keeping a food journal. These apps (Text4Diet [4], and MyFitnessPal [5]) maintain user privacy and are highly personalized, but do not provide any automated functionalities. The second group of apps, crowd-sourcing based semi-automatic, (Google's proposed app im2calories [6]) requires input from a large set of users to identify food. The system gets increasingly accurate as more users provide manual input. These apps can provide real time feedback, but they require users to share their food intake information. The third category of apps, personalized semi-automatic, (TECH [3]) requires the user to outline the food parts on a plate and also choose top 10 categories of food to help the identification process. These apps require manual intervention, but allow personalization and do not require food intake data to be made public. *MT-Diet* aims to have full automation in food segmentation, recognition, and quantity estimation. Recognition accuracy should be high to minimize user intervention, and segmentation performance should be high to improve recognition accuracy. Hence, these two processes are crucial for dietary monitoring However, there are major challenges:

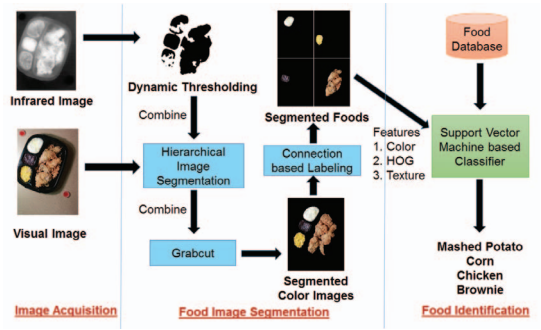


Fig. 1: *MT-Diet* food segmentation and identification.

a) Inaccurate segmentation - The average accuracy in the current state-of-the-art for automatically segmenting different food portions using a color image is below 63 % [7]. Further, when different food items are mixed it is nearly impossible to distinguish them using just the visual image of the food plate. In *MT-Diet*, we propose a *dynamic thermal thresholding (DTT)* algorithm to eliminate background and plate information from the thermal and color image and the *Region of Foods* (ROF) algorithm to accurately extract portions of food on both thermal and color image. By combining the color and thermal image, the *ROF* algorithm extracts food portions that are ignored by thermal thresholding due to low temperature difference from the plate, extracts food portions with same color as the plate by considering the thermal differences, and refines the food segments by eliminating plate portions that are close to food boundaries and heated due to conduction. Our results on a database of 80 types of cooked food, show that combination of

We are thankful to Dr. Petteri Nurmi for shepherding our paper and Dr. Meg Bruening for her insights on diet monitoring. This project is partially funded by NSF IIS 1116385 and NIBIB EB019202.

978-1-4673-8779-8/16/$31.00 ©2016 IEEE

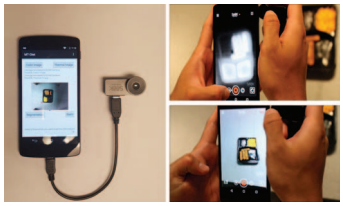

Fig. 2: Prototype of *MT-Diet* application using Nexus 5 and Seek thermal sensor.

thermal and visual images increases the segmentation accuracy to 97.5%. This can increase recognition accuracy leading to a substantial reduction of user intervention in diet monitoring.

b) Inaccurate identification - Visual image processing has been successful in identifying raw food such as fruits with nearly 100% accuracy. However, for cooked food, identification techniques from visual images have an accuracy of only 63% (Table I). *MT-Diet* augments machine learning classifiers with a thermal map of food plate. We propose *feature fusion and dimensionality reduction* techniques in improving accuracy and execution time of support vector machine based food classification. Experiments show that inclusion of food plate thermal map increases identification accuracy to 88.93%. This high accuracy can enable *MT-Diet* operation with only one click, significantly increasing its user friendliness.

## II. Related Work

Table I shows that: a) camera based diet monitoring systems have only used images in the visible spectrum, b) semi-auto segmentation with guidance from the user have a better accuracy than automatic algorithms, and c) automated identification have good accuracy for raw food but have low accuracy for cooked food. *MT-Diet* has the following advantages:

***MT-Diet* is independent of plate shape or color:** Two different types of attempts at food plate separation are traditionally pursued: a) assumption of a circular plate [9] allows the usage of Hough transform based edge detection to identify the plate boundary. This approach can only remove background and does not separate food from the plate. b) assumption of a white plate [12] such that any other color is food. But extraction of food items having same color as the plate can be difficult. Typically the above two methods work well for raw food, but not for cooked food (Table I). *MT-Diet* combines color and thermal images to segment food portions irrespective of plate color or shape.

***MT-Diet* improves identification using thermal images:** Food identification is typically performed using machine learning algorithms, which are trained to recognize certain identifying features of a food image such as statistical measures of color, texture, and edges. Experiments conducted using raw fruits suggest that existing machine learning methods have a high identification accuracy (97%), as also observed in past research [13]. However, with cooked food, the accuracy reduces significantly mostly due to two reasons: a) lack of well defined edges, and b) food items may have the same color as the plate. *MT-Diet* has an identification accuracy of 89% (25% increase) using information from the infrared spectrum, and optimizing the feature set using fusion techniques.

## III. Problem Definition and System Architecture

*Definition 1:* **Inputs: a) color image from a smartphone camera, and b) thermal image from an infrared camera.**

**Platform: a) A smartphone interfaced with a thermal camera, and b) reliable connection of the smartphone with a cloud server.**

**Assumptions:**
**a) Food temperature $\gg$ Plate temperature;**
**b) Plate temperature > Background temperature;**
**c) The plate is not overflowing with food; and**
**d) A database of food items is prepared offline and available to *MT-Diet*.**

**Outputs:**
**a) Food type in the plate; and**
**b) Estimation of ratio of carbohydrates, fat, fiber, and cholesterol.**

Assumptions (a) and (b) are important for the operation of *MT-Diet* and are observed to hold for meals cooked according to guidelines by U.S. Department of Health [14]. A rigorous analysis based on thermodynamics of the food and the plate is provided in the extended version of the paper [15].

**System architecture:** *MT-Diet* captures images of a food plate in both infrared and visual spectra through a thermal camera interfaced with an Android smartphone and the inbuilt camera (See Fig. 1 for the system architecture and Fig. 2 for the prototype configuration). The thermal image is used to segment food portions based on temperature differences along with the Grabcut method [16]. *MT-Diet* then analyzes the color histogram of each segment to determine the area covered by each food item. The food segments from both the thermal and color images are used to extract three features: a) Red Green Blue (RGB) color map, b) histogram of oriented gradients (HOG) [17], and c) texture information [18]. These features are then provided as input to a Support Vector Machine (SVM) based classifier to match to a pre-existing food database and extract the type of food [9], [12]. Hence, given a food plate, with minimal manual intervention a user can potentially obtain information such as type of food, calorie count, and percentage of carbohydrate, fat, fibre or cholesterol. Details of MT-Diet implementation [19] are skipped for the sake of brevity. A demonstration is also available online [20].

## IV. Food Segmentation

*Food segmentation process takes the color and thermal image of served hot food and plate in a background and outputs: a) number of different food items on the plate, and b) cropped region of each unique food item.*

The color image and thermal image are inputs to the food segmentation process. (detailed algorithm is provided in the extended version [15]). In the thermal image each pixel intensity is directly proportional to the temperature of the pixel. The output of the segmentation process is an image with only food pixels, excluding background and the plate. The first step in the algorithm is to find the edge image from the color image. A combination of three algorithms, Global Probability of Boundary, Oriented Watershed Transform, and Ultrametric Contour Map [21] is used to find the edges of plate, background and food portions. Using this edge information the Hierarchical Image Segmentation ($HIS$) [21] algorithm segments the color image and provides different labels to each segment. Traditionally, diet monitoring systems require manual intervention to increase accuracy of segments. However, *MT-Diet*, uses thermal image in the $DTT$ algorithm to improve segmentation accuracy without user inputs.

TABLE I: Camera based approaches that use images in the visible spectrum.

| Group / Methods | UEC [8] | DCVER [9] | He, Y. et al. [10] | Yang, S. et al. [11] | U of Bern [12] | MT-Diet |
|---|---|---|---|---|---|---|
| Identification Accuracy | 22 % | Single raw food 99% | 34% - 63 % | 78% | 87% | 88.93% |
| Segmentation Auto-level | Semi-Auto | Semi-Auto | Auto | Not Require | Semi-Auto | Auto |
| Identification Auto-level | Semi-Auto | Auto | Auto | Auto | Auto | Auto |
| Food Type | 100 Japanese foods | 39 raw food | 96 American food | Fast food | Six food items | 33 cooked food items |
| Plate Type | No assumptions | White round dish | White Round dish | Variable | White Round dish | No assumptions |

### A. Dynamic Thermal Thresholding (DTT)

Differences in temperature in a thermal image can be utilized to distinguish food on a plate with the same color. In practice, each food plate may not be heated to the same temperature. Hence, across gray scale thermal image, the intensity of food plate and background will vary, but there is a consistent trend that the food temperature is greater than food plate temperature and the plate temperature is higher than the background temperature.

The first step of $DTT$ is to seek the temperature of the background pixels. Based on the properties of the thermal camera any intensity value less than 150 is considered as a background pixel. The thermal image with background cancellation is denoted by $rBP$. The next step in DTT, is to find the plate temperature. In the background eliminated image, DTT searches for pixels with the highest difference in gray scale intensity from its neighbors. Hence for each pixel $rBP(i, j)$, it considers a $3 \times 3$ window ($W(i, j)$) as shown in Equation 1 to generate a difference matrix ($diff\_mat(i, j)$). The members of the $diff\_mat$ are the difference between maximum and minimum element in each $W$. If $W(i, j)$ has a background pixel, then $diff\_mat(i, j)$ is set to zero.

$$W(i,j) = \begin{bmatrix} \text{rBP (i-1 , j-1)} & \text{rBP (i-1 , j)} & \text{rBP (i-1 , j+1)} \\ \text{rBP (i , j-1)} & \text{rBP (i , j)} & \text{rBP (i , j+1)} \\ \text{rBP (i+1 , j-1)} & \text{rBP (i+1 , j)} & \text{rBP (i+1 , j+1)} \end{bmatrix},$$

$$diff\_mat(i,j) = \begin{cases} 0, & \text{if } Min(W(i,j)) = 0 \\ Max(W(i,j)) - Min(W(i,j)), & \text{else .} \end{cases} \quad (1)$$

The $(x, y)$ position with the maximum value in $diff\_mat$ represents a window $W(x, y)$ that has both food portions and plate. The median in this window $W(x, y)$ is considered as a threshold gray scale intensity value such that any pixel with intensity greater than this threshold can be classified as food. This method works if $T - T_p$ is more than twice the thermal sensor sensitivity. For the seek thermal camera [22], the sensitivity is around $0.5°C$. From our thermodynamic analysis (in extended version [15]), we see that $T - T_p > 20°C$ even after 15 min of wait. Hence, this technique is likely to work for cooked food. Although $DTT$ successfully removes the plate and background pixels, it does not segment individual food items. Using the Region of Foods ($ROF$) algorithm, we combine the color and thermal images to segment individual food items without user intervention.

### B. Region of Foods (ROF)

ROF algorithm tackles three problems as shown in Fig. 3.

**Case 1: Missing labels -** The $HIS$ generates only segments, it does not specify whether it is a food portion or not.

*Solution:* We combine the $HIS$ with the $DTT$, which has already separated the food portions from the plate and the background based on the thermal threshold. The $HIS$ and the $DTT$ segmentation portions are compared with respect to pixel indexes. If the indexes of each $HIS$ segmentation portion match with indexes of food portion as identified in $DTT$, the $HIS$ segmentation portion becomes candidates of $ROF$. This step outputs background segments and those that are either food or plate (Case 1 in Fig. 3).

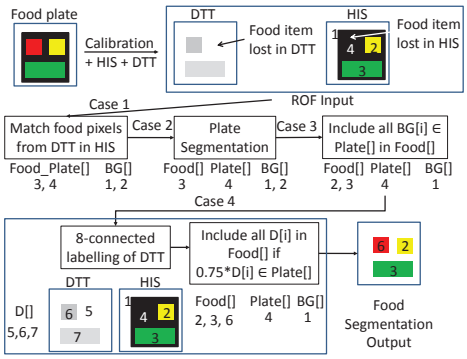


Fig. 3: The ROF algorithm showing four cases that might cause errors and their corresponding solutions.

**Case 2: Wrong labeling -** Plate portions near food items may get heated enough to be included in the output of the $DTT$ algorithm. In such cases, the $HIS$ segment, which corresponds to the plate area, may be wrongly classified as food (Fig. 4).

*Solution:* Case 1 has isolated the background portion of $eImg$. We scan the edge image starting from the four corner faces of the image to identify four corner pixels of plate. If three or more pixels amongst these have the same label numbers as assigned by the $HIS$ algorithm, the corresponding $HIS$ segment is considered as plate. The assumption is that the plate is not overflowing with food. The output of this case is segments which are only food items, segments which have majority of plate, and background segment (Case 2 in Fig. 3).

**Case 3: Missing food items in Thermal image-** Food portions which are not sufficiently heated may be removed from the output of DTT algorithm as shown in Fig. 4.

*Solution:* To solve this problem, we utilize the assumption that all food portions are contained within the plate. For each background segment in $HIS$ output if the median pixel falls within the plate segment of $HIS$ algorithm, we consider that background segment as food.

**Case 4: Missing food items in color image -** Food items, which have the same color as the plate are eliminated in the $HIS$ output as shown in Fig. 4.

*Solution:* This problem can be solved using the thermal image, because even if the food color is the same as the plate, the food temperature will be higher than the plate. To retrieve the such food portions, the segmented thermal image is labeled using

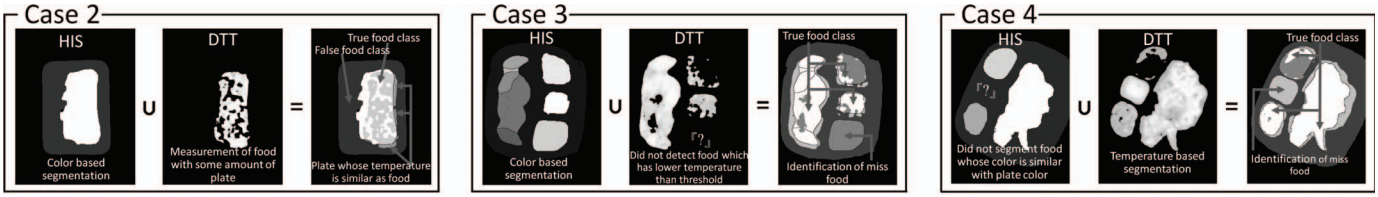


Fig. 4: Case 2: Elimination of plate portions that are heated to nearly similar temperatures as food. Case 3: Recovery of food portions that are not sufficiently heated. Case 4: Recovery of food portions with the same color as the plate.

the 8-connected component labeling [23]. For each labeled segment in *DTT* if over 75% of the connected component falls inside the plate segment in *HIS*, the *DTT* labeled portion is considered as a candidate *ROF*.

As a result of the four above-mentioned solutions, we obtain the candidates of *ROF*, however these candidates still have noises because *HIS* is an approximate segmentation method. To get accurate food portions, we use the Grabcut algorithm [16]. Given a selection of potential object and background, Grabcut provides more accurate object boundary based on the color distributions of the object and background. Typically, GrabCut is used in visual image based food segmentation in a semi-automatic setting, where the user is asked to select potential food portions also noted as *region of interest*. *MT-Diet* considers *ROF*s as region of interest in GrabCut and hence eliminates the need for user intervention.

## V. Food Identification

*Food identification process takes a food segment obtained from food segmentation as input and outputs the type of food items on the plate.*

### A. Feature Extraction

We consider three features: color, texture and HOG [17] from color and thermal images. First of all, RGB histogram is applied to extract color features. We generate 32 histogram bins of each color channel so that the dimension of the color feature vector is 32768 ($32\times32\times32$). The Gabor filter method is employed to extract the texture features. The segmented *ROF* portions are resized to a standard $400\times400$ image and five scales and eight orientations were considered. Each scaled and oriented image was then downsampled by a factor of 4 in each dimension. We extract HOG features [17], where each food image is divided into 16 windows and oriented gradients of the each window are calculated by 36 bin's histogram.

### B. Feature Fusion

The identification process requires a database of features of different types of food. The SVM is used to learn how to differentiate between different food types in the database. After the learning phase, the SVM is provided with the feature vector of an unregistered (not in the training database) food image. The SVM then attempts to classify the given input feature vector into a particular class using different distance functions known as the *kernel*.

Feature selection is an important trade-off between identification accuracy and response time of the dietary feedback.

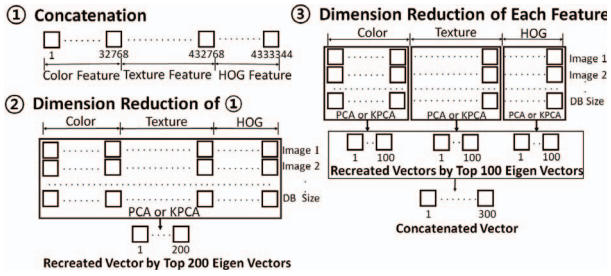


Fig. 5: Feature Fusion Methods.

Usage of a single feature such as either RGB or Gabor texture or HOG, may be computationally efficient, however, it has a drawback that the accuracy is greatly influenced by the nature of the database. For example, when the color feature is employed to classify, it is hard to classify foods with the same color like corn and cheese macaroni. To overcome such inaccuracies, one can fuse several features as shown in Fig. 5. However, feature size drastically increases resulting in higher computational time, and increasing response time of *MT-Diet*.

Further a simple concatenation of the color, texture and the HOG feature vector reuslts in high feature size (433344) hence increases the SVM execution time (Fig. 5). Therefore, we employ the dimensionality reduction techniques such as Principal Component Analysis (PCA) [24] and PCA with Gaussian Kernel Principal Component Analysis (KPCA) [25].

In our first feature reduction attempt, we consider the concatenated feature vector and perform PCA and KPCA, and select the top 200 eigen vectors from the feature matrix (Fig. 5). Using this eigen vector, we recreate the feature vector. Therefore, the feature dimension reduces to 200. The accuracy of the method, however, actually does not increase compared to the accuracy by using single feature vector because the color feature vector dominates the other feature vectors. In the second method, PCA and KPCA are applied separately to each feature before concatenating the feature vectors. This not only decreases feature vector size but also increase accuracy. The size of each feature is reduced to 100 and the whole feature vector size is 300 (Fig. 5).

## VI. Experimental Evaluation of MT-Diet

In this section, we evaluate MT-Diet food segmentation and identification method using experiments on cooked meals.

### A. Experimental Setup

In our experiments, we used 80 cooked and frozen dishes as the sample database. The food plates were de-frozen and

heated to recommended temperatures using a microwave. The 80 food plates consists of overall 33 different type of foods.

Each food plate had either single or multiple food items. The plates were of different shapes based on the number of food items available. The database consists of 244 *ROF* images from all 80 food plates.

To implement the SVM classifier we used the libsvm library [26]. The software supports kernels such as Linear, Polynomial (Poly), Radial Basis Function (RBF), and Sigmoid and k-fold cross validation. In our experiments, all kernels were evaluated with 5-fold cross validation with the three types of individual features and three feature fusion methods.

### B. Experimental Results

In this section, we consider four important metrics for evaluating MT-Diet: a) accuracy for food segmentation, b) accuracy of identification, c) execution time of food segmentation, and d) execution time of food identification. The accuracies are required to evaluate the usability of *MT-Diet* as a diet monitoring tool, and the execution times are required to evaluate the response time of dietary feedback.

*1) Food Segmentation Accuracy:* For segmentation, we focused on two issues: how well the background and plate pixels were removed and how well the multiple food portions were separated. Fig. 6 displays a sample output of our segmentation method for three images with mixed food (results for the whole database is available online [15]). Food segmentation error have to be computed with respect to human visual evaluation. Two error cases might exist: a) food segments have plate pixels in them, which can result in inaccurate feature extraction and reduce food identification accuracy, and b) mixed food items may not be separated.

For the first error case, we could visually identify two images out of 80 which have plate pixels included after the food segmentation process. In the first picture we have around 6.5% while the other picture has 0.8% plate pixels amongst food pixels. Hence the plate portions in *ROF*s are minimal and may not affect the identification accuracy.

For the second case, the total number of food items observed visually (244) was different from that obtained by *MT-Diet* (210). This is because the human observer could distinguish between mixed food items which *MT-Diet* failed. Food items such as rice mixed in with turkey gravy are mixed without any color edge and were counted as single item by *MT-Diet*. To overcome such errors, we inserted mixed food items in to our database as new data. For example, turkey with rice, was inserted as a single item. This reduced the error in food identification process since the counting errors were confined to only two food plate images out of 80, resulting in 97.5% accuracy but increased the execution time.

*2) Food Identification Accuracy:* Table II shows the accuracy and execution time for each type of feature fusion method and SVM kernel. An extended version of this table is available online [15]. We consider three features and their combinations: Gabor (texture), HOG, and RGB histogram. For each individual methods, the fusion method is labeled as $NTH$ in Table II. For each feature we used dimensionality reduction (denoted as $D \cdot R$ in Table II) using two types of PCA, with linear and Gaussian kernel function.

For each combination we consider two types of fusion as discussed in Section V-B and Fig. 5: a) simple concatenation, and then dimensionality reduction using PCA or KPCA, labeled as *Concatenate* and b) dimensionality reduction of each feature and then concatenation, labeled as $Separate$. Four SVM kernels were used including linear, polynomial, RBF, and sigmoid. The fusion of color and texture or color and HOG with $Separate$, $KPCA$, and RBF kernel has the highest average accuracy (88.93%).

TABLE II: Food identification accuracy and execution time.

| Feature | Fusion Method | D · R Method | Accuracy (%) Kernel | Time (s) Kernel | Feature Size |
|---|---|---|---|---|---|
| RGB | NTH | KPCA | 88.11 (Sigmoid) | 0.57 | 100 |
| **HOG & RGB** | Concatenate | PCA | 84.43 (RBF) | 1.71 | 200 |
| | **Separate** | **KPCA** | **88.93 (RBF)** | **0.70** | **200** |
| RGB & Gabor | Concatenate | KPCA | 43.85 (RBF) | 6.38 | 200 |
| | Separate | KPCA | 88.93 (RBF) | 5.62 | 200 |
| All | Concatenate | KPCA | 43.85 (RBF) | 6.56 | 200 |
| | Separate | KPCA | 87.70 (RBF) | 5.43 | 300 |

*3) Food Segmentation Execution Time:* In food segmentation, there are four main tasks: $DTT$, $HIS$, $ROF$, and *Grabcut*. According to Table III, the execution time of $HIS$ (83.08%) and *Grabcut* (12.78%) occupied 95.86% of execution time.

TABLE III: Execution time (s) of food segmentation.

| | Min | Median | Max | Sum | Avg | STD |
|---|---|---|---|---|---|---|
| HIS | 89.08 | 92.89 | 97.50 | 7403.90 | 92.55 | 1.66 |
| DTT | 2.83 | 3.13 | 3.90 | 256.12 | 3.20 | 0.24 |
| ROF | 0.39 | 0.78 | 1.01 | 62.77 | 0.78 | 0.09 |
| Grabcut | 2.40 | 14.24 | 41.15 | 1138.97 | 13.48 | 7.63 |
| Total | 95.51 | 111.59 | 145.00 | 8911.79 | 111.64 | 7.71 |

*4) Food Identification Execution Time:* To evaluate the food identification execution time, we considered two main tasks: Dimensionality reduction and SVM training depending on the kernel types. According to Table II, the average execution time of the *RBF* kernel is the highest and the average execution time of the *Linear* kernel is the lowest. Performing $KPCA$ was better than $PCA$ with respect to the dimensionality reduction execution time. Also, the execution time of PCA and KPCA increases with respect to the feature vector size. The SVM training time dominated the food identification execution time. Further, the Dimensionality Reduction method is necessary to not only improve accuracy but also reduce the execution time.

*5) Full system performance evaluation:* The most computationally expensive operation in MT-Diet is the food segmentation taking almost $100s$. The data transfer takes $5s$ while the food identification method takes around $5s$ (Table II). Hence, the response time of *MT-Diet* i.e., the time between capturing a picture and getting the food type information is $110s$. In terms of memory, communication bandwidth and energy consumption requirements, *MT-Diet* is lightweight. The app takes around 940 KB in the smartphone SD card. During operation it transmits the color and thermal image to the server, a total of 975 KB. The server transmits the segmented food portions back to the mobile phone, a total of 236 KB. Average power consumption of the *MT-Diet* application was 17 mW.

## VII. Discussion

**Usability of MT-Diet for diet monitoring:** MT-Diet requires images of food plate in the thermal and visual spectrum. Smart

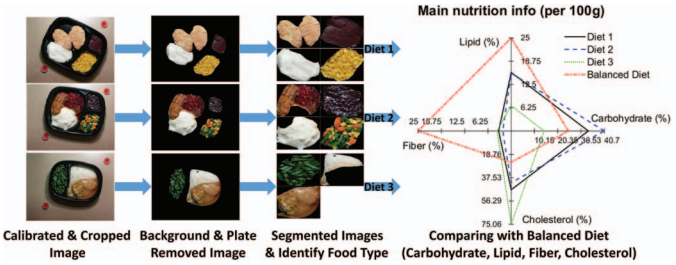


Fig. 6: Segmentation, identification, and diet recommendation.

watches in the pervasive computing domain already interface cameras in wristbands. Thermal cameras can be interfaced easily through micro-usb ports as shown in Fig. 2 or even embedded in the hardware. Image capture can be invoked by recognizing sequence of hand gestures.

Once the images are captured, the smartphone can be used as a hub for data communication and computation. The smartphone may choose to implement food segmentation and identification or may chose to offload the implementation to the cloud server. To implement in the cloud server, the smartphone has to send both the thermal and visual image to the server. On the other hand, for offline implementation in the smartphone, it has to download a learned classification machine, trained using a sample database.

The result of the identification procedure is a set of labeled image areas each corresponding to a specific food item on the plate. The output can be used for several purposes including calorie intake estimation, balanced diet evaluation, or checking conformity to a specific type of diet. One of the several possible outputs is discussed next.

**Balanced diet recommendation:** The output of the food identification process is a set of image areas with identified food items. We assume that the plate has uniform depth and hence a ratio of the surface area multiplied by the density of the food items (obtained from USDA website) gives the ratio of weight of different food items. We then normalize the amount of each food item on the plate by considering the total food weight to be 100 grams. For each food item we derive the amount of carbohydrates, lipids, fibers, and cholesterol content for the normalized weight using the statistics per 100 gram of each food item from USDA. For a balanced diet the carbohydrates, lipids, fibers, and cholesterol must have equal weights in the food plate. A balanced diet should have each component at 25%. Given a food plate MT-Diet shows how far it is from a balanced diet in the form of a spider chart (Figure 6) and what component of the diet should be changed in order to make it a balanced diet.

## VIII. Conclusions

We introduced, *MT-Diet* an automatic diet monitoring system that interfaces thermal sensor with smartphone camera to provide accurate food identification. MT-Diet improves the accuracy of automated food identification to 88.93%, a 25% increase with respect to competitive techniques. The usage of thermal information not only helps in identifying the food portions in the plate but also in recognizing the type of food in a time efficient manner, within 2 mins of taking a picture. *MT-Diet* is an user-friendly diet monitoring application that is expected to promote healthy eating habits.

## References


[1] J. Milazzo, P. Bagade, A. Banerjee, and S. K. S. Gupta, "bhealthy: A physiological feedback-based mobile wellness application suite," in *Proceedings of the 4th Conference on Wireless Health*, ser. WH '13. New York, NY, USA: ACM, 2013, pp. 14:1–14:2.

[2] A. Banerjee and S. Gupta, "Analysis of smart mobile applications for healthcare under dynamic context changes," *Mobile Computing, IEEE Transactions on*, vol. 14, no. 5, pp. 904–919, May 2015.

[3] H. Henriksson, S. E. Bonn, A. Bergström, K. Bälter, O. Bälter, C. Delisle, E. Forsum, and M. Löf, "A new mobile phone-based tool for assessing energy and certain food intakes in young children: A validation study," *JMIR mHealth and uHealth*, vol. 3, no. 2, 2015.

[4] J. R. Shapiro, T. Koro, N. Doran, S. Thompson, J. F. Sallis, K. Calfas, and K. Patrick, "Text4diet: a randomized controlled study using text messaging for weight loss behaviors," *Preventive medicine*, vol. 55, no. 5, pp. 412–417, 2012.

[5] [Online]. Available: https://www.myfitnesspal.com/.

[6] G. Burgett, "Google researcher develops im2calorie, an ai that can count calories in food photos," June 2015, accessed: 2015-07-10. [Online]. Available: http://www.imaging-resource.com/news/2015/06/08/im2calorie-a-google-backed-ai-that-can-count-calories-in-food-photos

[7] M.-Y. Chen, Y.-H. Yang, C.-J. Ho, S.-H. Wang, S.-M. Liu, E. Chang, C.-H. Yeh, and M. Ouhyoung, "Automatic chinese food identification and quantity estimation," in *SIGGRAPH Asia*. ACM, 2012, p. 29.

[8] Y. Kawano and K. Yanai, "Real-time mobile food recognition system," in *Computer Vision and Pattern Recognition Workshops (CVPRW),*. IEEE, 2013, pp. 1–7.

[9] P. Pouladzadeh, S. Shirmohammadi, A. Bakirov, A. Bulut, and A. Yassine, "Cloud-based svm for food categorization," *Multimedia Tools and Applications*, pp. 1–18, 2014.

[10] Y. He, C. Xu, N. Khanna, C. J. Boushey, and E. J. Delp, "Food image analysis: Segmentation, identification and weight estimation," in *Multimedia and Expo (ICME), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1–6.

[11] S. Yang, M. Chen, D. Pomerleau, and R. Sukthankar, "Food recognition using statistics of pairwise local features," in *Computer Vision and Pattern Recognition (CVPR),*. IEEE, 2010, pp. 2249–2256.

[12] M. Anthimopoulos, J. Dehais, P. Diem, and S. Mougiakakou, "Segmentation and recognition of multi-food meal images for carbohydrate counting," in *Bioinformatics and Bioengineering (BIBE), 13th International Conference on*. IEEE, 2013, pp. 1–4.

[13] P. Pouladzadeh, P. Kuhad, S. V. B. Peddi, A. Yassine, and S. Shirmohammadi, "Mobile cloud based food calorie measurement," in *Multimedia and Expo Workshops (ICMEW)*. IEEE, 2014, pp. 1–6.

[14] [Online]. Available: https://www.health.ny.gov/environmental/indoors/food_safety/coolheat.htm.

[15] Online Appendix. [Online]. Available: https://impact.asu.edu/PerCom2016_ver5.pdf.

[16] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics (TOG)*, vol. 23, no. 3, pp. 309–314, 2004.

[17] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition. CVPR.*, vol. 1. IEEE, 2005, pp. 886–893.

[18] B. S. Manjunath and W.-Y. Ma, "Texture features for browsing and retrieval of image data," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, no. 8, pp. 837–842, 1996.

[19] J. Lee, A. Banerjee, and S. K. S. Gupta, "Mt-diet demo: Demonstration of automated smartphone based diet assessment system," in *Pervasive Computing and Communications (PerCom)*. IEEE, 2016.

[20] [Online]. Available: https://www.youtube.com/watch?v=En8iyJ5JSI4.

[21] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 5, pp. 898–916, 2011.

[22] Accessed: 2015-07-10. [Online]. Available: http://www.thermal.com/.

[23] R. M. Haralock and L. G. Shapiro, *Computer and robot vision*. Addison-Wesley Longman Publishing Co., Inc., 1991.

[24] I. Jolliffe, *Principal component analysis*. Wiley Online Library, 2002.

[25] B. Schölkopf, A. Smola, and K.-R. Müller, "Kernel principal component analysis," in *Artificial Neural Networks ICANN*. Springer, 1997, pp. 583–588.

[26] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.