A Mobile User Interface for Semi-automatic Extraction of Food Product Ingredient Lists

Tobias Leidinger, Lübomira Spassova, Andreas Arens, Norbert Rösch
CRP Henri Tudor, CR SANTEC, 29, avenue J.F. Kennedy, L-1855 Luxembourg
{tobias.leidinger, lubomira.spassova, andreas.arens, norbert.roesch}@tudor.lu

ABSTRACT

The availability of food ingredient information in digital form is a major factor in modern information systems related to diet management and health issues. Although ingredient information is printed on food product labels, corresponding digital data is rarely available for the public. In this article, we present the Mobile Food Information Scanner (MoFIS), a mobile user interface that has been designed to enable users to semi-automatically extract ingredient lists from food product packaging. The interface provides the possibility to photograph parts of the product label with a mobile phone camera. These are subsequently analyzed combining OCR approaches with domain-specific post-processing in order to automatically extract relevant information with a high degree of accuracy. To ensure the quality of the data intended to be used in health-related applications, the interface provides methods for user-assisted cross-checking and correction of the automatically recognized results. As we aim at enhancing both the data quantity and quality of digitally available food product information, we placed special emphasis on fast handling, flexibility and simplicity of the user interface.

Author Keywords

Mobile user interface; information extraction; food product information; optical character recognition; data acquisition.

ACM Classification Keywords

H.5.2. [Information Interfaces and Presentation]: User Interfaces – Graphical user interfaces, Interaction styles

INTRODUCTION & RELATED WORK

Ingredient information about food products can be interesting for different groups of people due to either ethical or health reasons, such as finding organic or vegan ingredients, or filtering out products unsuitable for allergy sufferers or diabetes patients. Although ingredient information is printed on food product labels, corresponding digital data is rarely available for the public. Various online food databases provide ingredient data; most of them are user-maintained, such as Codecheck.info, Barcoo.com, Fddb.info, Das-ist-drin.de or Wikifood.eu, which has been developed and is maintained by the CRP Henri Tudor. Many platforms also provide interfaces for mobile devices, so that food information becomes more easily accessible. Most databases rely on the participation of volunteers and therefore offer interfaces for users to add new products or edit product information. However, manual entry of food information is tedious and time consuming, which restricts the growth of the corresponding databases.

According to a 2005 WHO report [6] and Robertson et al. [14], the attitude of European consumers is changing towards the intensified consumption of healthy food. In order to enable users to take informed decisions concerning the healthiness of food, extensive food information platforms are needed. Research on how to encourage users to voluntarily provide digital data have resulted in game-like approaches, for example the mobile game Product Empire by Budde and Michahelles [2], where users can build virtual empires by uploading product information. Still, to the best of our knowledge, none of the approaches that involve user input of digital food data offer methods for automated image-based data extraction of ingredient lists, although the data is available, printed on the product labels. In addition, food product labels change frequently as indicated by Arens et al. [1], so that it is important to keep the data up-to-date.

Image processing and optical character recognition (OCR) have been in the focus of research for many years, and corresponding commercial tools as well as open-source solutions are available for desktop computers. There are mobile OCR tools using server-based methods to process images captured by mobile devices, such as Google Goggles[1] or the ABBYY Business Card Reader[2]. For translation purposes, mobile apps for instant recognition and translation of written text are available, e.g., Word Lens[3] or TranslatAR [5]. Laine and Nevalainen describe a theoretical approach to how OCR can be performed directly on mobile devices [9], and there is furthermore at least one prototype implementing the Tesseract OCR engine[4] for Android devices[5]. Most applications of OCR tools require scanned text documents that are of high quality and have a clear contrast between text and background to produce good recognition results. Ingredient lists,

IUI 2013 Workshop: Interacting with Smart Objects, March 18, 2013, Santa Monica,CA, USA

[1] http://www.google.com/mobile/goggles/
[2] http://www.abbyy.com/bcr/
[3] http://questvisual.com/us/
[4] http://code.google.com/p/tesseract-ocr/
[5] https://github.com/rmtheis/android-ocr

however, have various appearances like different shapes of product packaging, varying foreground and background colors, glossy surfaces or irregular background patterns. In our research setting, the pictures are assumed to be taken with a low resolution mobile phone camera. Taghva and Stofsky point out that OCR errors vary from device to device, from document to document and from font to font [17]. This can induce problems with word boundaries and typographical mistakes. In their article, Taghva and Stofsky present a spelling correction system for the Emacs platform on desktop computers. However, OCR input and error correction on a mobile phone require rather multi-modal approaches. The advantages of multi-modal systems for error correction over uni-modal ones are discussed in a user study by Suhm et al. [16]. Dumas et al. describe the differences between classical graphical user interfaces (GUI) and multi-modal ones (MUI) [3]. The semi-structured nature of ingredient lists makes full automation extremely challenging, although the domain vocabulary of food product data is rather restricted. To compensate for weaknesses of the OCR and post-processing systems, the user has to input unknown words or unrecognized parts using other text input methods, like the mobile phone's keyboard. In this context, Kristensson discusses challenges for intelligent text entry methods [8]. She states that "text entry methods need to be easy to learn and provide effective means of correcting mistakes".

Most references concerning multi-modal interfaces focus on speech recognition together with correction methods. In the last decades, several user interfaces for speech recognition error correction have been presented. The systems interpret spoken language and the interfaces allow the user to correct the result, for example using pens, touch screens or keyboards at mobile phones or 3D gestures at Kinect-based game consoles. Some of the techniques encompass word confusion networks, which show different candidates for each recognized word that can be replaced [12, 18], some show interfaces that allow the user to select sentence or word alternatives [4], and others use the dasher interface, which allows users to navigate through nested graphical boxes in order to select subsequent characters [7]. In the food-related domain, Puri et al. propose an approach to refining the results of an image-based food recognizer by allowing the user to list out each of the food items present in a picture using spoken utterances [13]. Although the project uses disambiguation of recognition results through incorporation of input from different modalities (image and speech), the system by Puri et al. does not offer any further opportunity for error handling through the user.

The present paper introduces the Mobile Food Information Scanner MoFIS – a user interface for mobile devices that enables semi-automatic OCR-based information extraction from food product packaging.

MOBILE USER INTERFACE

The MoFIS interface aims at enabling users to add product information to a food product database in an effortless way with their mobile phones, in order to enhance both the data quantity and quality of digitally available food product information. We have implemented a prototype that provides possibilities for scanning product barcodes, taking pictures of product packaging, marking and tagging regions of interest (ROIs) and cross-checking and correcting product information with a special emphasis on fast handling, flexibility and simplicity of the user interface.

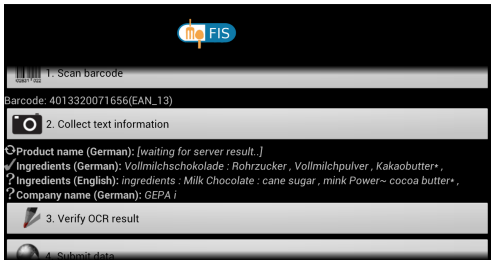

Figure 1: Main menu of the MoFIS application.

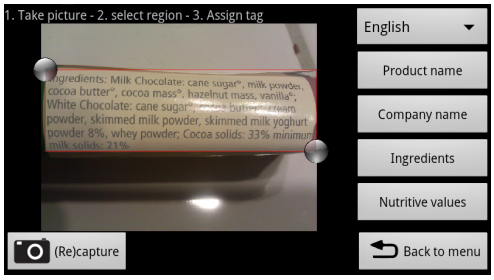

Figure 2: Image capturing interface with selected ROI.

Main menu

Using four buttons, the user can start the different tasks of the data acquisition process (1. barcode scanning, 2. taking pictures to collect text information, 3. correcting and verifying the OCR result and 4. submitting data). Products are identified by their EAN codes using the ZXING barcode scanner library[6].

Collecting information

For providing product information, the MoFIS app offers the user the possibility to take several pictures of the food product packaging in order to capture all areas containing relevant information, which is automatically extracted using OCR. The data preview in the main menu adapts whenever the user provides or edits food-related information or when a provided image has been successfully interpreted by the system. Small status icons to the left of the respective preview line visualize the current state of each data item (Figure 1). In each of the pictures, the user can mark several regions of interest (ROIs). Furthermore, the language and the type (product name, company name, ingredients, nutritive values) of each fragment can be specified (Figure 2).

Result confirmation

The user is presented an overview of all information that has been extracted and interpreted by the OCR engine so far. It

[6]http://code.google.com/p/zxing

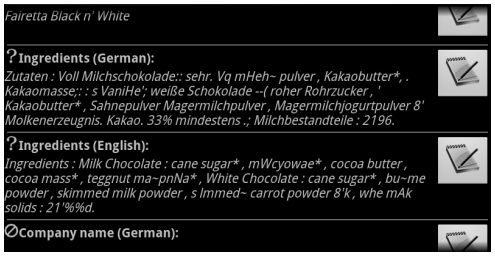

Figure 3: Overview of OCR results.

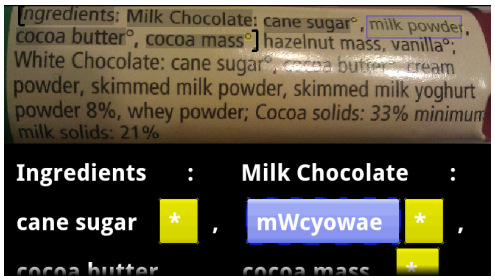

Figure 4: Ingredient list overview.

shows the status of the result for each ROI, so that the user can identify possible errors at a glance (Figure 3). The edit button opens a view for further analysis and error correction. In order to facilitate the user's interaction during the confirmation task, the corresponding original picture has been integrated in different views of the user interface. The aim of this approach is to keep the user focus on the screen and, in this way, to reduce the probability of errors.

Due to the complex semi-structured nature of ingredient lists and the significance of their content, the interface provides particular user support in the task of finding possible errors in the corresponding OCR results and offers different possibilities for error correction. An *ingredient list overview* enables cross-checking by showing a section of the original image containing the ingredient list in the upper half and the corresponding text candidates in the bottom half of the screen (Figure 4). The two parts of the view are synchronized in such a way that the image shows only a section corresponding to the visible candidates, and scrolling of the candidates shifts the image accordingly. Furthermore, the exact section of candidates currently visible in the lower view is marked with brackets in the original picture in the upper view. The overview initially shows the best automatically retrieved candidates according to a comparison with an ingredient dictionary. Candidates with a low confidence score are highlighted with a yellow or a red background, based on the error distance between the best dictionary match and the OCR result. The overview provides the opportunities to *remove* wrong candidates, *edit* minor errors or manually *insert* parts where the OCR entirely failed. A long press on a candidate opens a context menu with the corresponding options. The user can additionally double tap on every candidate to see a detailed *error correction view* presenting several candidates for each position in the ingredient list provided by the OCR post-processing engine. In order

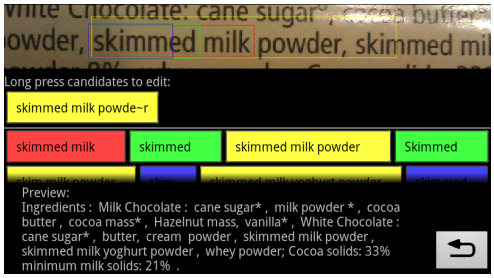

Figure 5: Detailed error correction view.

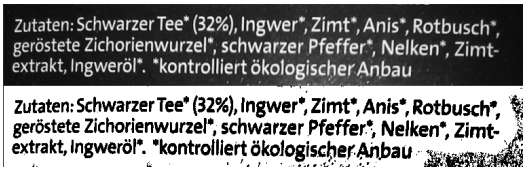

Figure 6: Sample of the pre-processing: original image and binarized version.

to compensate for text segmentation problems and to enable the matching of composed ingredient terms, candidates with different text lengths are considered, which are presented in different colors in the correction view. The user can decide which candidate fits best to the original text marked with a box of a corresponding color (Figure 5). If the correct candidate is not suggested but a close match, the user can long press on a candidate in the correction view to change the proposed text. In this way, the user can benefit from the OCR results and thus speed up the data acquisition process even if the recognition results might not be perfectly correct. After the user has made a selection, the subsequent word candidates are automatically adapted by the selection engine.

IMPLEMENTATION

Although modern smartphones become more and more powerful, we followed the approach of the majority of mobile OCR clients and send the captured pictures to a server for pre-processing, OCR and post-processing. The implementation of the user interface is so far limited to Android devices.

Text extraction

The open-source OCR tool Cuneiform[7] is used to extract the ingredient information from the source image. Along with the text data, Cuneiform provides the bounding boxes of recognized letters, which are used to layout the preview components in the user interface. The OCR software is used as a black box, so that improvements are limited to appropriate pre- and post-processing methods.

Pre-processing

In order to achieve reasonable results, the OCR software requires binarized images. A good binarization algorithm for text images amplifies the contrast between text and background. Especially in the case of poor quality pictures of product packaging, pre-processing is an essential step. For

[7] https://launchpad.net/cuneiform-linux

this project, we apply a modified version of Sauvola's binarization method [15, 10]. Figure 6 shows an example of the result of the pre-processing algorithm.

Dictionary matching

Our post-processing method can compensate for OCR mistakes while trying to preserve the structure of ingredient lists. Some ambiguities can not be resolved completely automatically, and thus, the user interface offers alternatives, i.e., the possibility to manually change the preselected candidates.

Structure of ingredient lists

Generally, ingredients are free-text phrases, separated by commas or semicolons. Brackets enclose nested sub-lists of ingredients, and special patterns, like declarations of weight or percentages, can be appended to an ingredient. As Taghva and Stofsky state in [17], OCR systems can misinterpret letters, numbers, dividers and spaces, so that detecting word boundaries is not always feasible. Therefore, reconstruction of the correct ingredient list is only possible with a semi-automated approach involving human assistance.

Entity matching

In this work, the existing WikiFood ingredient list database was used in order to extract a corpus of food ingredient terms and corresponding term frequencies. We use the concept of bi-grams to enable access to potential candidates in constant time. For this purpose, the OCR result is parsed and split into individual words. Subsequently, all word entities are collected and matched against the ingredient corpus. In addition, a sliding window with a certain window length is used to combine adjacent ingredients in order to improve the matching of corrupt words and to be able to propose composed ingredient terms in the user interface. On the one hand, this creates overheads as the matching algorithm is executed more often, but on the other hand this approach significantly improves the result of the post-processing. In the matching phase, all candidate alternatives are ordered by their matching quality as compared to the OCR result, taking into account text edit distance, term frequency and term length. The edit distance is calculated using the Levenshtein distance [11] between the OCR result and the candidates. The output of this algorithm is a sorted list of candidates for the start position (offset) of every entity of the OCR result. Candidates at the same offset can span different words or letters and may overlap with neighboring candidates. Depending on the suggestions, it is up to the user to select composed ingredient terms or several individual words one after the other. The algorithm outputs a maximum of 10 best-ranked candidates for every offset position in order to limit network traffic and to filter out inappropriate candidates. The preliminary ingredient list is initially automatically composed of the best-ranked consecutive, non-overlapping candidates.

PERFORMANCE

To test the recognition rate of the MoFIS engine, we chose 20 random food products covering all different packing characteristics in shape (cylindrical, rectangular, uneven) and color (various text colors, background colors, transparent plastic foil). We took pictures of the ingredient lists with a standard mobile phone camera (5MP, auto-focus) indoors under normal daylight conditions, using the automatically triggered flash light. We counted (a) the number of ingredients that were correctly selected initially, without user interaction, (b) the number of candidates that could be found, but had to be selected from the list of alternatives and (c) the number of candidates that had to be input manually[8]. In average, (a) 90.38% of the candidates were recognised correctly, (b) 4.08% could be chosen from suggested alternatives and (c) only 5.54% had to be inserted manually.

CONCLUSION

In this work, we have presented the MoFIS system, consisting of an OCR server and a mobile user interface that can be used to capture pictures with the mobile device, process the pictures on the server and let the user validate and correct the results directly on the phone. Using the MoFIS interface, usually only little effort is necessary to accomplish the correct results. The system can be used to automatically extract and validate ingredient information from food product packaging using a mobile phone, which is the first such attempt to the best of our knowledge. Compared to basic OCR approaches, this leads to more complete and accurate data with only small additional effort.

We claim that this method provides an enormous advantage for user-maintained food databases compared to traditional text input. In the MoFIS system, most of the text is extracted automatically, so that only little user interaction is necessary. Finding errors – and especially correcting them – is supported by the user interface by presenting both original preview and user input simultaneously. As most modern mobile phones have a camera of sufficient quality and as it is possible to run the OCR and post-processing on a server in reasonable time, this mechanism can provide an adequate alternative to current food-related data acquisition approaches, e.g., through web platforms.

In our future research, we plan to evaluate the MoFIS system by conducting user-centered studies and to adapt and extend the system based on the results and the user feedback gathered in the course of these studies.

REFERENCES

1. Arens-Volland, A., Rosch, N., Feidert, F., Herbst, R., and Mosges, R. Change frequency of ingredient descriptions and free-of labels of food items concern food allergy sufferers. *Special Issue: Abstracts of the XXIX EAACI Congress of the European Academy of Allergy and Clinical Immunology, London, UK 65*, s92 (2010), 394.

2. Budde, A., and Michahelles, F. Product Empire - Serious play with barcodes. *Internet of Things (IOT), 2010* (2010), 1–7.

[8]We only considered textual ingredient phrases for this evaluation step, ignoring dividers, special signs and numbers when counting the number of words.

3. Dumas, B., Lalanne, D., and Oviatt, S. Human Machine Interaction. Springer-Verlag, Berlin, Heidelberg, 2009, ch. Multimodal, 3–26.

4. Feld, M., Momtazi, S., Freigang, F., Klakow, D., and Müller, C. Mobile texting: can post-ASR correction solve the issues? an experimental study on gain vs. costs. In *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*, IUI '12, ACM (New York, NY, USA, 2012), 37–40.

5. Fragoso, V., Gauglitz, S., Zamora, S., Kleban, J., and Turk, M. TranslatAR: A mobile augmented reality translator. In *IEEE Workshop on Applications of Computer Vision (WACV), 2011* (2011), 497–502.

6. Hayn, D. *Ernährungswende: Trends und Entwicklungen von Ernährung im Alltag (Food Change: Trends and developments in nutrition in everyday life)*, vol. 2. Inst. für sozial-ökologische Forschung (ISOE), 2005.

7. Hoste, L., Dumas, B., and Signer, B. SpeeG: a multimodal speech- and gesture-based text input solution. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, AVI '12, ACM (New York, NY, USA, 2012), 156–163.

8. Kristensson, P. O. Five Challenges for Intelligent Text Entry Methods. *AI Magazine 30*, 4 (2009), 85–94.

9. Laine, M., and Nevalainen, O. A standalone ocr system for mobile cameraphones. In *PIMRC*, IEEE (2006), 1–5.

10. Leidinger, T., Arens-Volland, A., Krüger, A., and Rösch, N. Enabling optical character recognition (OCR) for multi-coloured pictures. In *Proceedings of the ImageJ User and Developer Conference, Edition 1*, ISBN: 2-919941-18-6 (2012).

11. Levenshtein, V. I. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady 10* (Feb. 1966), 707+.

12. Ogata, J., and Goto, M. Speech Repair: Quick Error Correction Just by Using Selection Operation for Speech Input Interface. In *Proc. Eurospeech05* (2005), 133–136.

13. Puri, M., Zhu, Z., Yu, Q., Divakaran, A., and Sawhney, H. Recognition and volume estimation of food intake using a mobile device. In *Workshop on Applications of Computer Vision (WACV), 2009* (2009), 1–8.

14. Robertson, A., Tirado, C., Lobstein, T., Jermini, M., Knai, C., Jensen, J. H., Luzzi, A. F., and James, W. P. T. Food and health in Europe: a new basis for action. Tech. rep., pp 7-90, 2004.

15. Sauvola, J., and Pietikinen, M. Adaptive document image binarization. *PATTERN RECOGNITION 33* (2000), 225–236.

16. Suhm, B., Myers, B., and Waibel, A. Model-based and empirical evaluation of multimodal interactive error correction. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, CHI '99, ACM (New York, NY, USA, 1999), 584–591.

17. Taghva, K., and Stofsky, E. OCRSpell: an interactive spelling correction system for OCR errors in text. *International Journal of Document Analysis and Recognition 3* (2001), 2001.

18. Vertanen, K., and Kristensson, P. O. Parakeet: a continuous speech recognition system for mobile touch-screen devices. In *Proceedings of the 14th international conference on Intelligent user interfaces*, IUI '09, ACM (New York, NY, USA, 2009), 237–246.