



# FussyFood Melhorado

Isabel Souza de Carvalho (Graduanda em Ciência da Computação), Luiza Helena Vieira (Graduanda em Ciência da Computação)  
Ciência da Computação - Visão Computacional  
Universidade Federal do Maranhão – UFMA  
Emails: bel.souza07@gmail.com , luizahelenavieira92@gmail.com

## 1. INTRODUÇÃO

Em 2015, Sameep Bagadia e Rohit Mundra desenvolveram um aplicativo chamado FussyFood que tem como objetivo o reconhecimento de ingredientes a partir de imagens de embalagens de alimentos. A aplicação auxilia pessoas com restrições nutricionais ou alergias alimentares na escolha

## 2. OBJETIVO

O objetivo deste trabalho é apresentar algumas propostas para incrementar o funcionamento do aplicativo FussyFood.

## 3. PROPOSTAS APRESENTADAS

As propostas para melhoria da metodologia utilizada pelos autores citados não visam a modificação do fluxo de processamento mostrado na Figura 1, mas algumas mudanças na forma como cada passo do fluxo é executado. Alguns dos passos do fluxo de processamento serão analisados e terão propostas para melhoria em sua execução, conforme relatado nas seções a seguir.

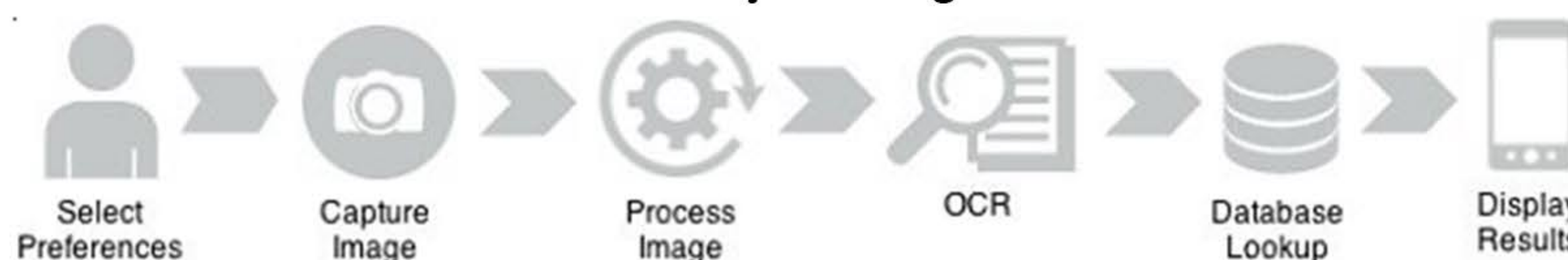


Figura 1: Fluxo de Processamento no FussyFood

Fonte: Bagadia e Mundra (2015)

### 3.1 Seleção de Preferências

A primeira etapa do processamento no aplicativo FussyFood é a seleção de preferências de acordo com as restrições alimentares do usuário. O aplicativo contém 6 (seis) opções de restrições. A proposta para essa etapa é a separação das preferências alimentares que são analisadas em conjunto, de modo que cada restrição alimentar seja tratada separadamente.

### 3.2 Captura da Imagem

A segunda etapa consiste na captura da imagem. O aplicativo FussyFood foi projetado para capturar imagens de superfícies planas. A proposta para essa etapa diz respeito à captura de imagens de superfícies cilíndricas, pois há diversas embalagens de alimentos em formato de cilindro e pessoas com algum tipo de restrição alimentar não se limitam em consumir produtos com embalagem plana. Em imagens onde as linhas de texto são curvadas e as palavras próximas das extremidades laterais são mais estreitas, os sistemas OCR apresentarão falhas. Bieniecki (et al, 2007) propõem uma forma para processar imagens não lineares. O primeiro passo para a correção é traçar curvas que delimitem as bordas superior e inferior da área onde encontra-se o texto.

Encontradas as curvas superior e inferior, o procedimento de correção é executado. Cada é transformada em uma linha perpendicular usando as equações de interpolação, como resultado é criada a imagem retangular

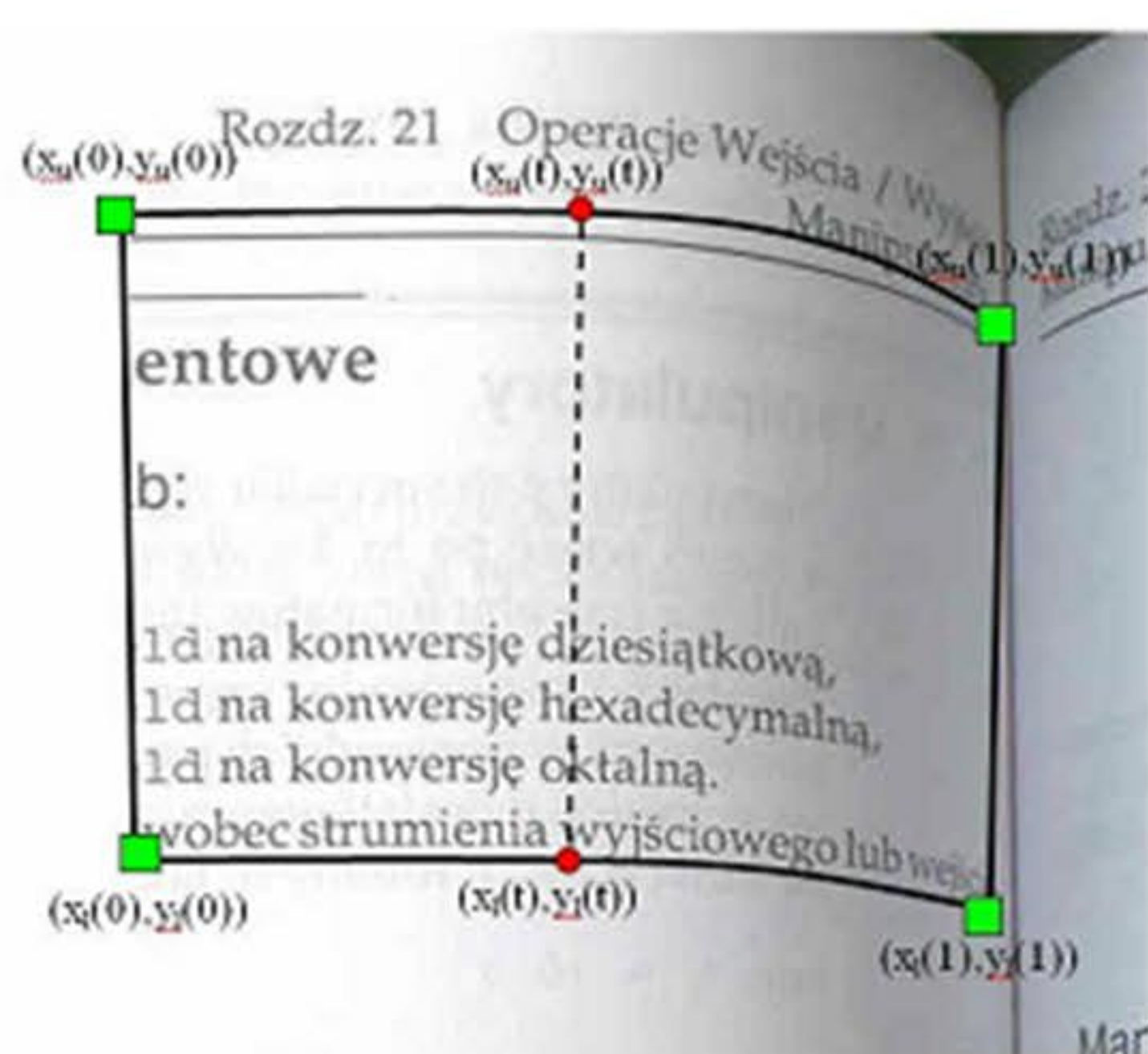


Figura 2: Encontrando as curvas de delimitação para a área de interesse.  
Fonte: adaptado de Bieniecki (et al, 2007)

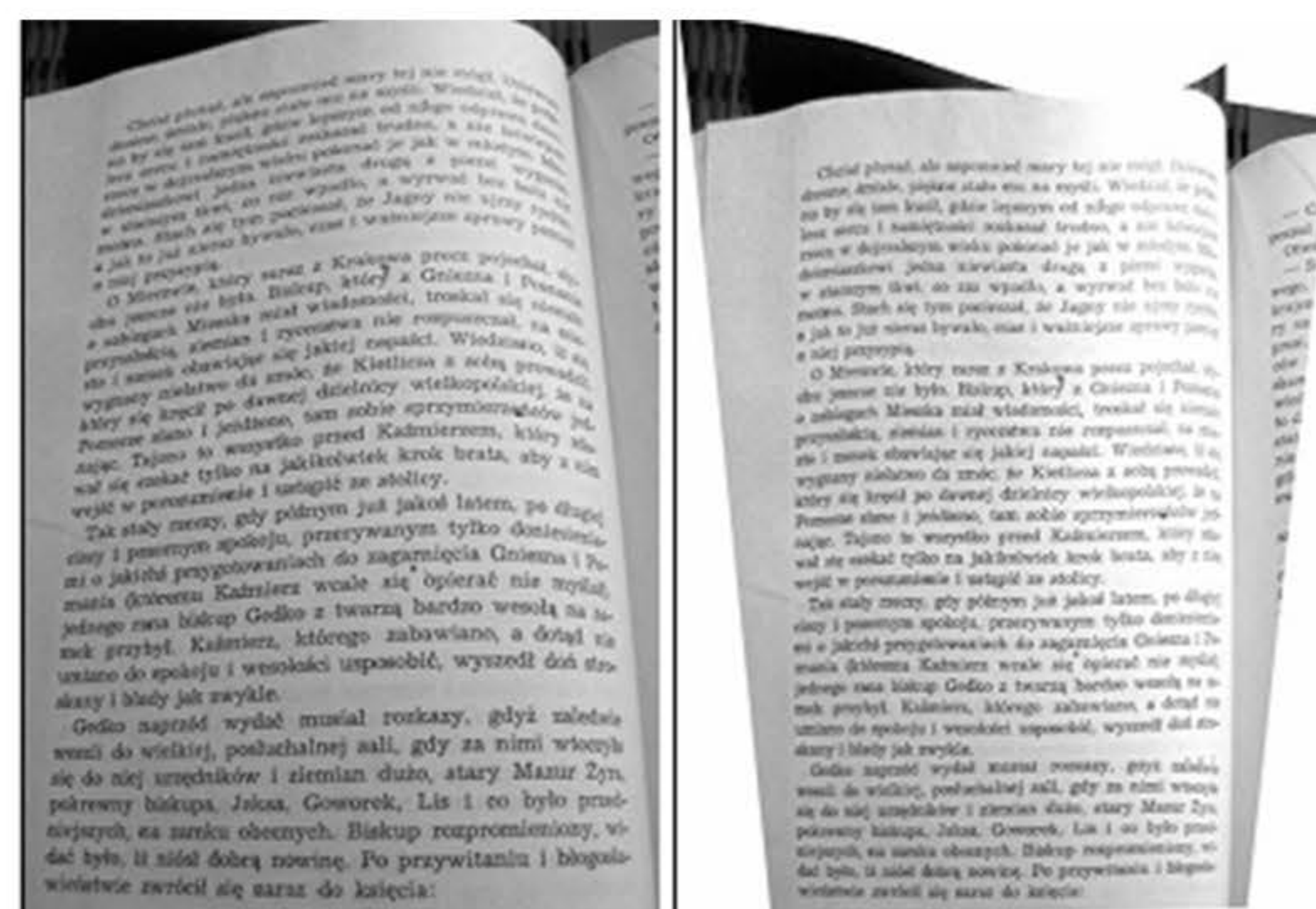


Figura 3: Resultado da restauração de uma imagem.  
Fonte: adaptado de Bieniecki (et al, 2007)

### 3.3 Processamento da imagem

Essa etapa seguirá a metodologia proposta por Bagadia e Mundra (2015). As alterações ocorrerão apenas em caso de tratamento de imagem de embalagens cilíndricas, se a imagem capturada pelo usuário for cilíndrica, ocorrerá primeiramente o processamento descrito na seção 2.2, com a aplicação da metodologia proposta por Bieniecki (et al, 2007). Capturada a imagem, serão aplicados os métodos utilizados no trabalho de Bagadia e Mundra (2015): o thresholding adaptativo local, a transformada probabilística de Hough, a remoção do background e a detecção de bounding box.

### 3.4 Reconhecimento óptico de caracteres e busca no banco de dados

Depois de todo processamento e tratamento da imagem, a parte a ser utilizada para assimilar as restrições consiste na parte dos ingredientes. Após separar as palavras que se referem aos ingredientes, o aplicativo faz uma busca no banco de dados e compara com as preferências do usuário.

A proposta de melhoria nessa etapa é o uso de um corretor ortográfico, para tratar nomes de ingredientes em que foi reconhecido com algum erro, como, por exemplo, choocolate. Não houve proposta de melhoria para a base de dados, pois os autores não especificaram que base de dados eles utilizaram na aplicação.

Como a aplicação, FussyFood, foi desenvolvida para Android, um corretor ortográfico a ser utilizado pode ser o framework desenvolvido pela plataforma, Spelling Checker Framework for Android. O corretor usa as palavras obtidas da imagem como entrada no framework, gerando assim uma série de sugestões de palavras corretas.

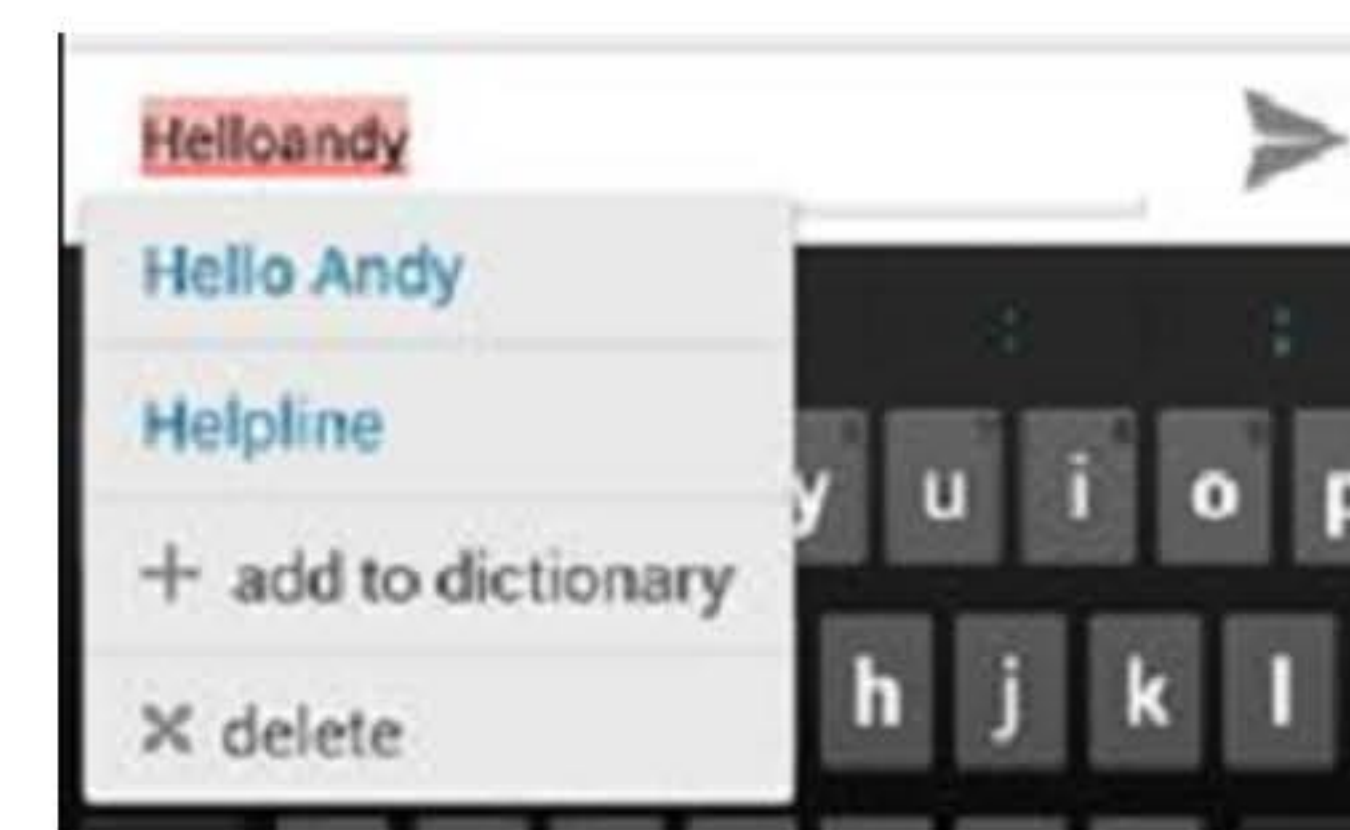


Figura 4: Spell checker em uso  
Fonte: Spelling Checker Framework Documentation (2016)

## 4. REFERÊNCIAS

- Bagadia, S. e Mundra, R. " FussyFood: An Application to Navigate Food Allergies Dietary Restrictions". Stanford University, California, 2015.
- Bieniecki, W., Grabowski, S and Rozenberg, W. "Image preprocessing for improving ocr accuracy," in Perspective Technologies and Methods in MEMS Design, 2007. mEMSTECH 2007. International Conference on, pp. 75–80, IEEE, 2007.
- Spelling Checker Framework Documentation. Disponível em < <http://developer.android.com/guide/topics/text/spellcheckerframework.html> > Acesso em: 31 de março. 2016