

Departamento de Sistemas e Computação – FURB
Curso de Ciência da Computação
Trabalho de Conclusão de Curso – 2014/1

Plantarum: uma aplicação Android para consultas de plantas

Acadêmico: Matheus Bortolon
matheusbortolon@gmail.com

Orientador: Prof. Aurélio Hoppe
aurelio.hoppe@gmail.com
<http://www.inf.furb.br/~aurelio/>

Grupo de Pesquisa em Computação
Gráfica, Processamento de Imagens e
Entretenimento Digital
<http://www.inf.furb.br/gcg>



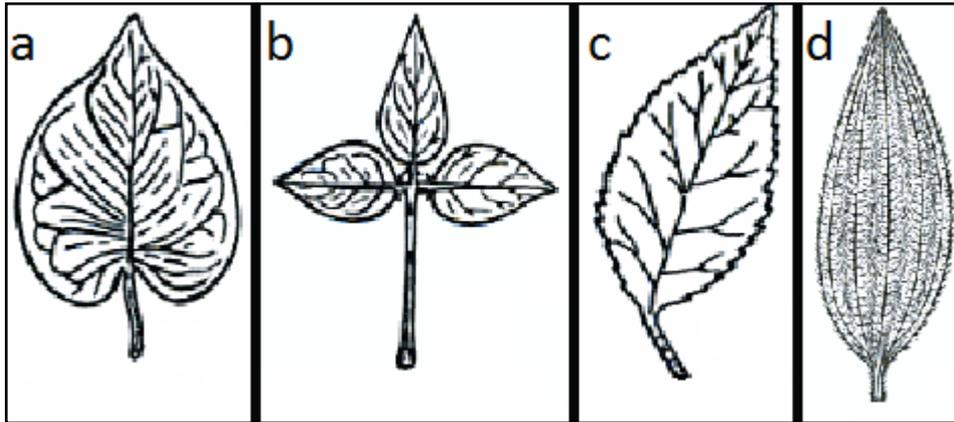
Roteiro

- Motivação
- Fundamentação
- Trabalhos relacionados
- Trabalho proposto
- Requisitos
- Desenvolvimento
- Experimentos
- Conclusão
- Limitações
- Extensões
- Demonstração

Motivação

- A taxonomia é uma tarefa realizada manualmente, devido a seu caráter especializado
- Tarefa torna-se mais complicada, devido à grande biodiversidade existente no Brasil
- Diferentemente de flores, folhas estão presentes em diversas espécies de plantas e possuem maior independência das estações do ano

Fundamentação



Tipos de folha

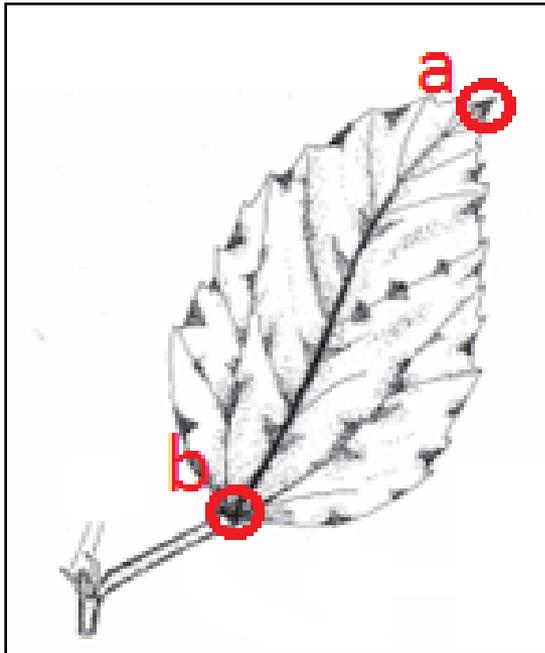
- a) Simples
- b) Composta
- c) Lisa
- d) Pilosa
- e) Discolor

API Plantarum

- A API Plantarum permite a classificação de plantas a partir da imagem digital de folhas
- Extrai da imagem a região correspondente à folha da planta
- Extrai descritores morfológicos, geométricos e características de cor das amostras
- Classifica as plantas dentre as espécies previamente registradas em uma base de dados

API Plantarum

- Na versão desenvolvida por Cassaniga além das características extraídas, a API considera a entrada do usuário para informação de pecíolo e ponta da folha



- a) Ponta
- b) Pecíolo

API Plantarum

- Foram incluídas na API três novas características a serem informadas pelo usuário a fim de cobrir variações morfológicas
 - Formato, onde é solicitado que o usuário informe se a folha é simples ou composta
 - Coloração, onde o usuário deve confirmar se a folha é discolor ou não
 - Textura, onde deve ser informado se a folha é pilosa ou lisa

API Plantarum

- Para a utilização das novas características foram criadas três novas classes na API, e foi alterada a estrutura de passagem de parâmetros em dois métodos
- As três classes estendem de `IFeature`, para possibilitar que sem nenhuma outra alteração fossem utilizadas pela rede neural para a classificação
- Para a persistência não foi preciso alterar a API devido ao fato de estender da interface `IFeature`

Trabalhos relacionados

- Foram estudados trabalhos correlatos que faziam classificação de espécies por meio de dispositivos móveis, são eles:
 - Leafsnap: A computer vision for automatic plant species identification. Desenvolvido por Kumar et al. em 2012.
 - Automatic plant leaf classification for a mobile field guide: an Android application. Desenvolvido por Knight, Painter e Potter em 2010
 - Shape based plant leaf classification system using Android. Desenvolvido por Sangle, Shisat e Bhosle em 2013.

Trabalhos relacionados

- O trabalho de Kumar et al. mostra o desenvolvimento de uma aplicação para IOS
- A aplicação é end-end, necessitando de um servidor para classificar as espécies fotografadas
- No trabalho é descrito que eles utilizam a silhueta para fazer a classificação das amostras
- Ao final do processo são retornadas as vinte espécies mais similares com a espécie submetida para classificar

Trabalhos relacionados

- Interfaces de exemplo do Leafsnap:

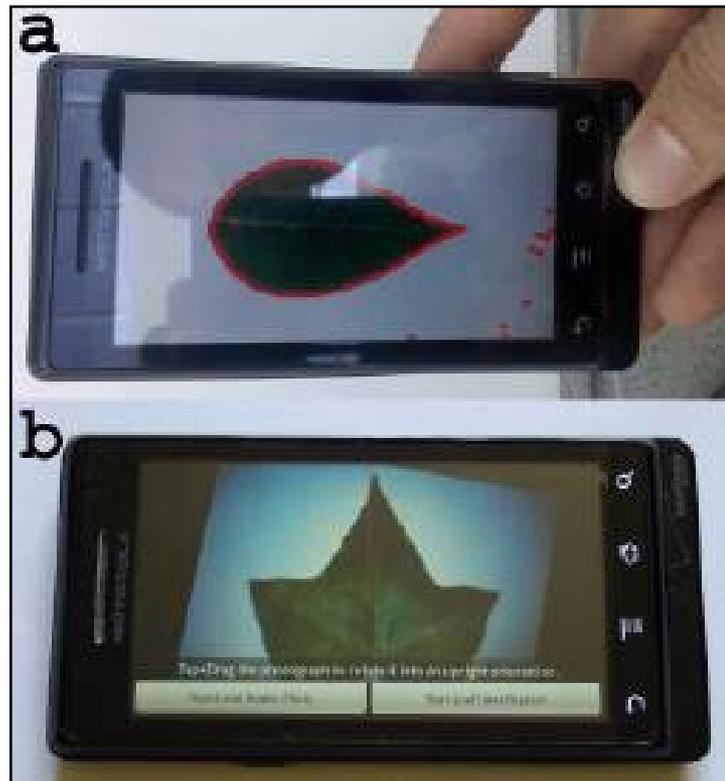


Trabalhos relacionados

- O trabalho de Knight, Painter e Potter mostra o desenvolvimento de uma aplicação para Android
- A aplicação computa localmente as características, não necessitando de um servidor para classificar
- No trabalho é descrito que eles utilizam a silhueta e descritores geométricos para fazer a classificação das amostras
- Ao final do processo é retornada apenas a espécie que possui maior similaridade com a amostra escolhida

Trabalhos relacionados

- Exemplos do trabalho de Knight, Painter e Potter:



Trabalhos relacionados

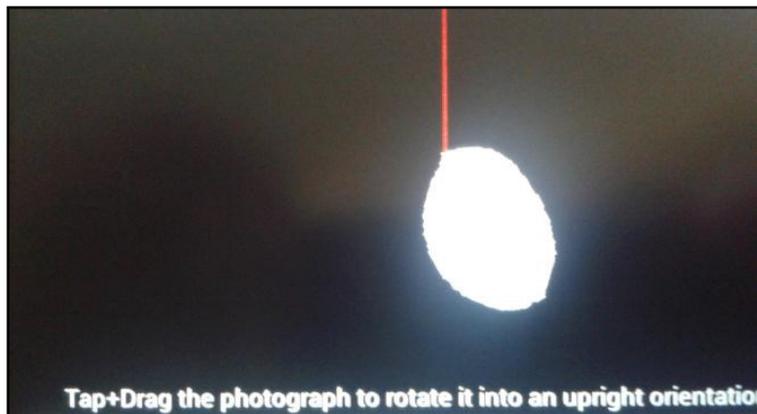
- O trabalho de Sangle, Shirsat e Bhosle também mostra o desenvolvimento de uma aplicação para Android
- Da mesma forma que Knight, Painter e Potter, a aplicação de Sangle, Shirsat e Bhosle computa localmente as características
- Eles utilizam a silhueta e descritores geométricos para fazer a classificação das amostras como Knight Painter e Potter
- Ao final do processo é exibido o nome da espécie que possui maior similaridade com a amostra escolhida

Trabalhos relacionados

- Exemplos do trabalho de Sangle Shirsat e Bhosle:



GroundNut	Distance: 0.8604571
Gulmohar	Distance: 0.9607355
Tulsi	Distance: 2.4664493
Ashoka	Distance: 2.031358
Mango	Distance: 1.9769436
Rose	Distance: 1.6749703



Trabalhos relacionados

Características / Trabalhos relacionados	Kumar et al. (2012)	Knight, Painter e Potter (2010)	Sangle, Shirsat e Bhosle (2013)
Coordenadas de GPS	Sim	Não	Não
Extração das características	Servidor	Dispositivo	Dispositivo
Método de entrada	Câmera do dispositivo móvel	Câmera do dispositivo móvel	Câmera do dispositivo móvel
Método de reconhecimento	Com auxílio do usuário	Automático	Automático
Plataforma mobile	iOS	Android	Android

Trabalho proposto

Disponibilizar uma aplicação para Android capaz de interagir com um *webservice* para efetuar o cadastro e classificação de espécies de plantas.

Objetivos:

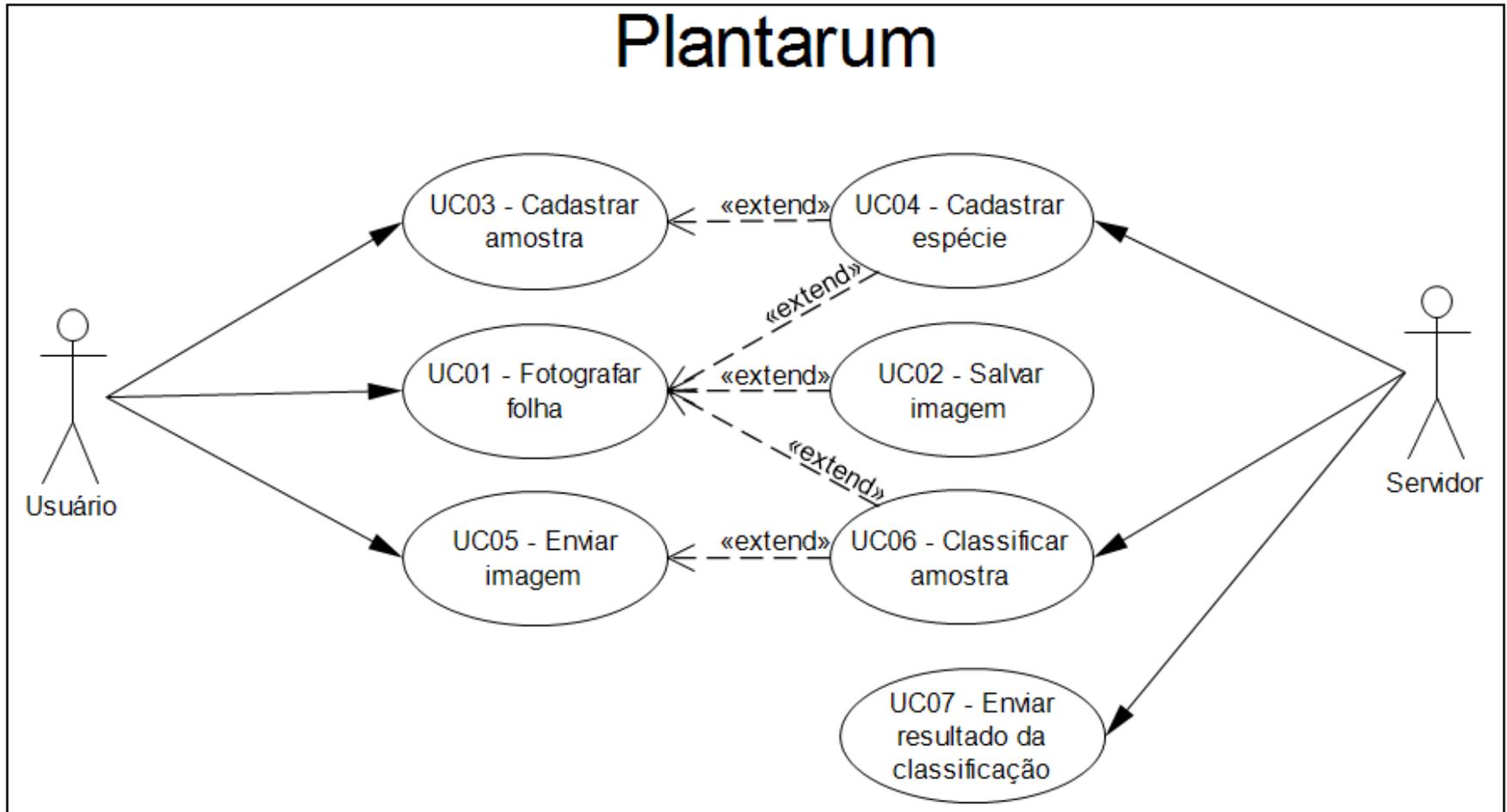
- Capturar imagens a partir da câmera do dispositivo móvel
- Prover um serviço WCF para realizar a troca de informações entre o dispositivo móvel e o API Plantarum
- Fornecer comunicação e armazenamento assíncronos, para que na indisponibilidade do servidor as informações possam ser armazenadas para envio posterior.

Requisitos

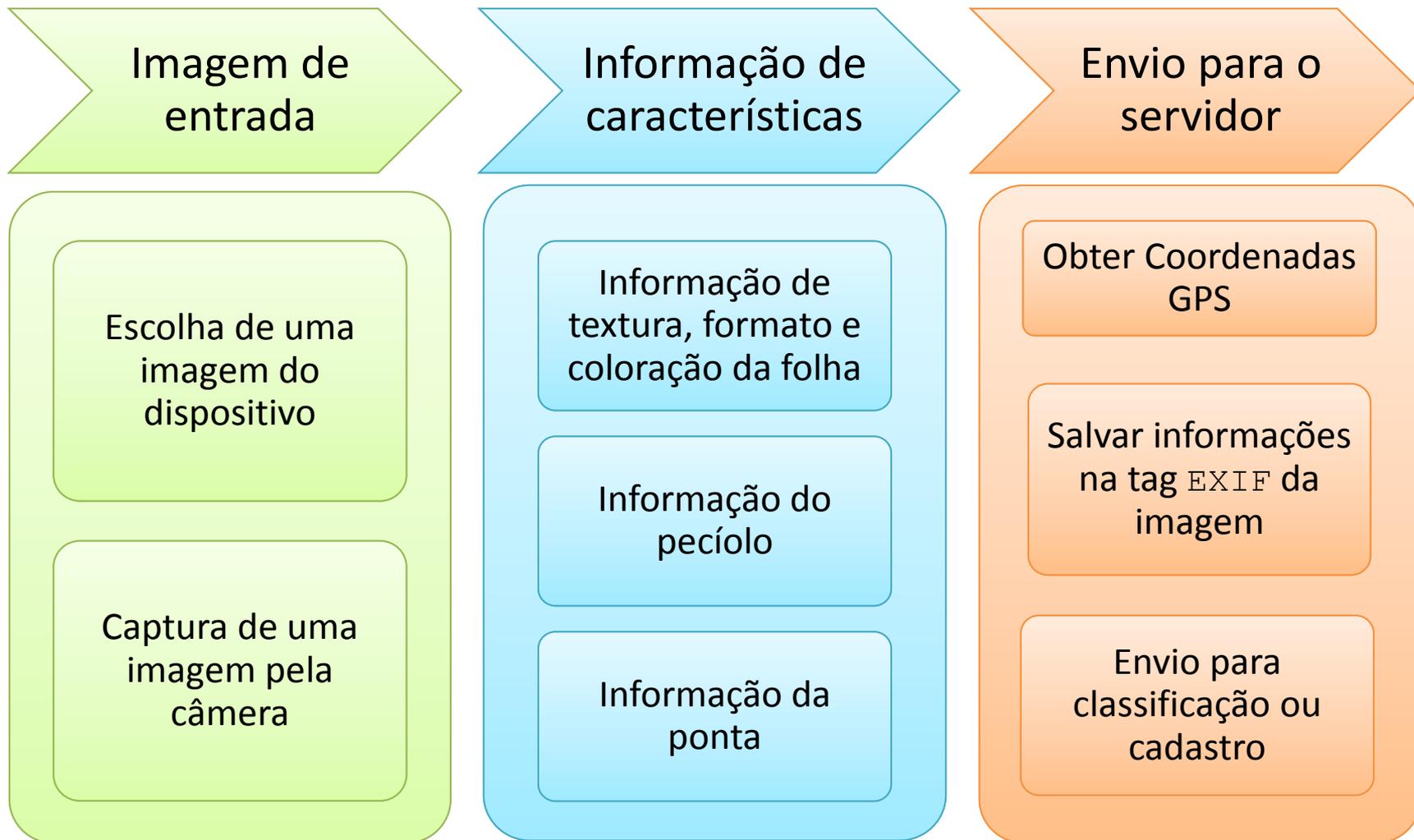
A seguir estão enumerados os principais requisitos funcionais do projeto:

- Permitir acesso a câmera com o intuito de fotografar uma amostra (RF)
- Possuir funcionalidade de envio ao servidor, para classificação e cadastro de amostras (RF)
- Permitir salvar no dispositivo funcionalidade de envio ao servidor para classificação e cadastro de amostras (RF)
- Possuir um serviço que utilize a API Plantarum como mecanismo de cadastro e classificação (RF)

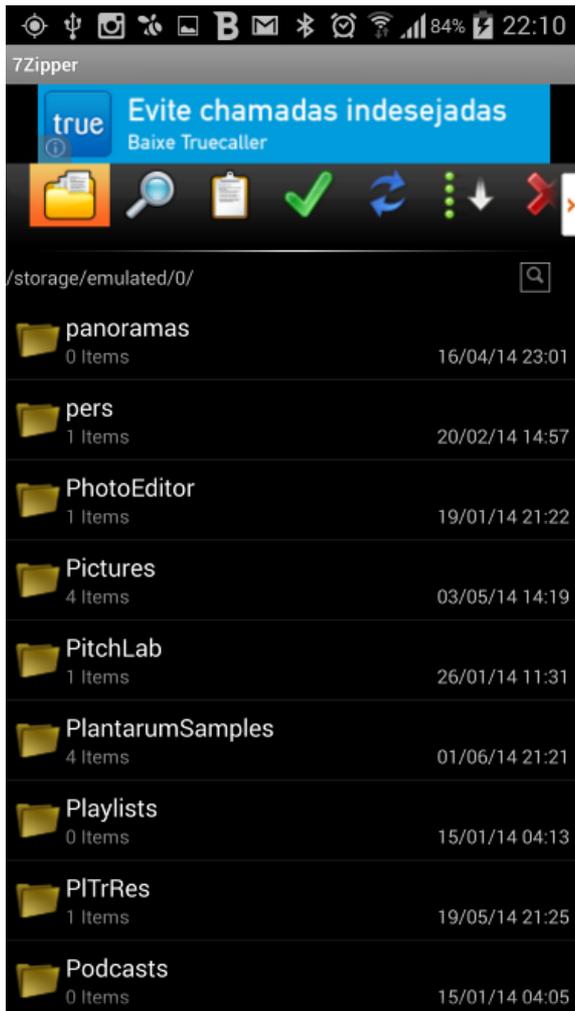
Caso de uso da aplicação



Etapas da classificação ou cadastro na aplicação cliente

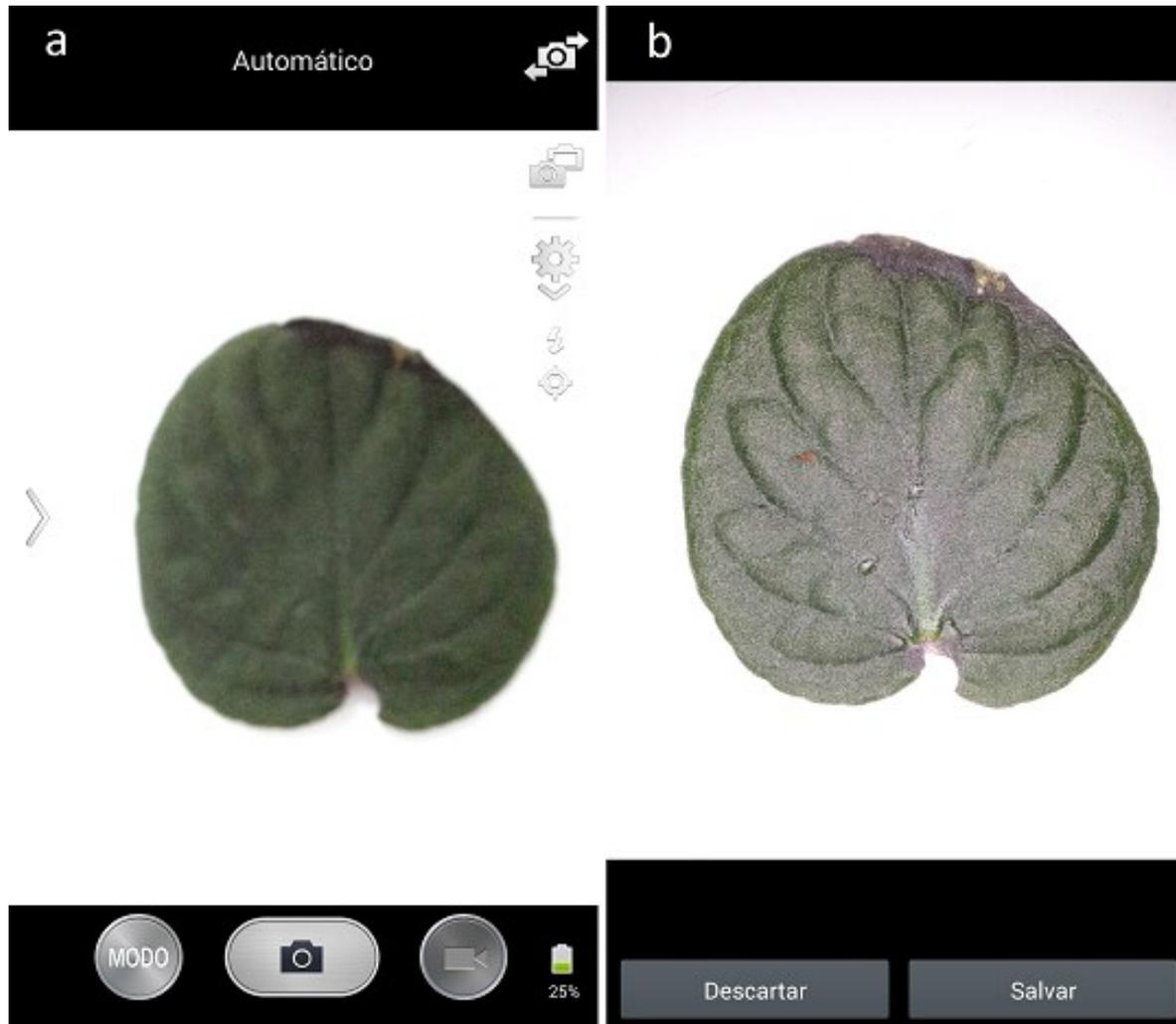


Entrada de dados – Selecionar arquivo



- Aplicação padrão de gerenciador de arquivos
- Copia o arquivo para um temporário onde serão gravadas as informações
- Verifica se já tem os detalhes na imagem
- Verifica se deve ir para o cadastro de amostras

Entrada de dados - Câmera



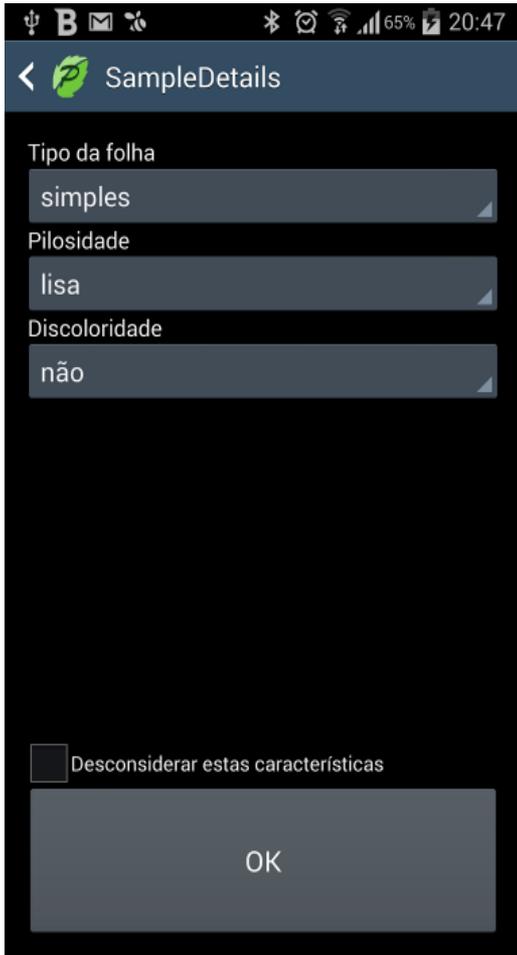
- a) Aplicação padrão de câmera
- b) Após à captura é exibida a imagem capturada

Entrada de dados - câmera

- Após retornar da câmera, é tratado o cancelamento da operação
- Atribuídas as informações de GPS
- Inicia a atividade para informação de características

```
if (resultCode != RESULT_CANCELED) {  
    if (!Exif.hasGPSTagsOnImage(Constants.IMAGE_PATH))  
        Exif.setGPSInfo(Constants.IMAGE_PATH, this);  
  
    startIntentSampleDetails();  
    return;  
}
```

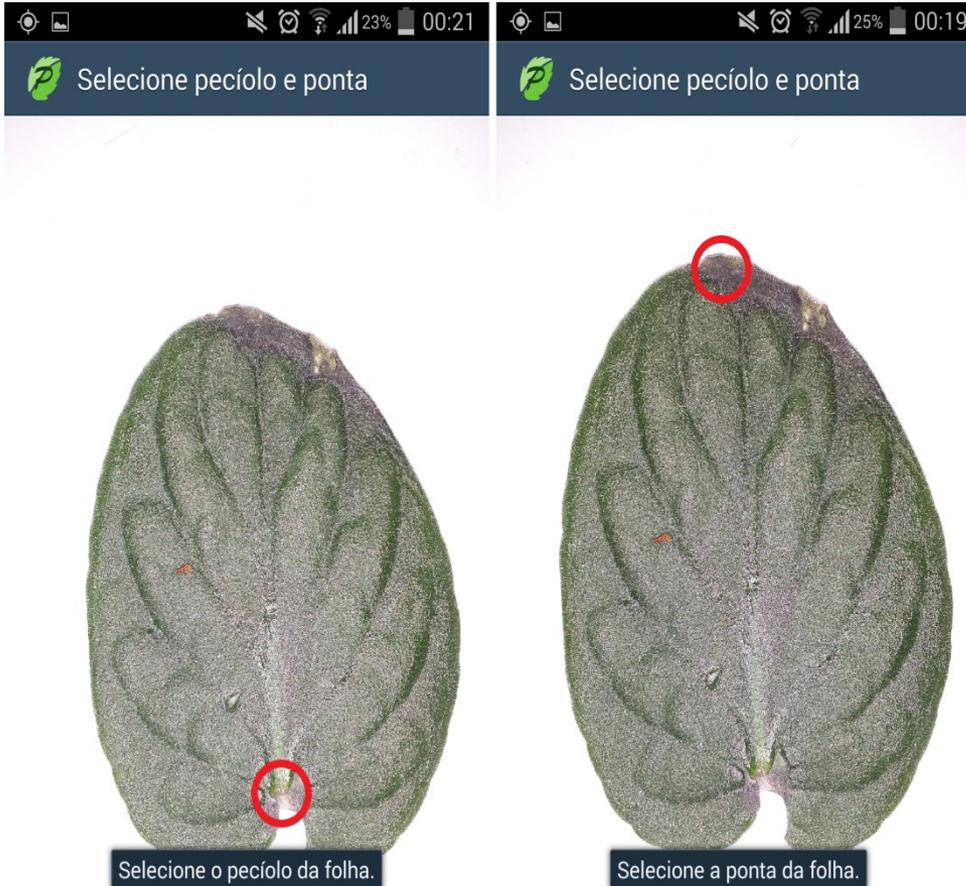
Informação de características – Novas características da API



The screenshot shows a mobile application interface with a dark theme. At the top, there is a status bar with various icons and the time 20:47. Below that is a header bar with a back arrow, a green leaf icon, and the text 'SampleDetails'. The main content area contains three dropdown menus: 'Tipo da folha' with 'simples' selected, 'Pilosidade' with 'lisa' selected, and 'Discoloridade' with 'não' selected. At the bottom, there is a checkbox labeled 'Desconsiderar estas características' which is currently unchecked, and a large grey button labeled 'OK'.

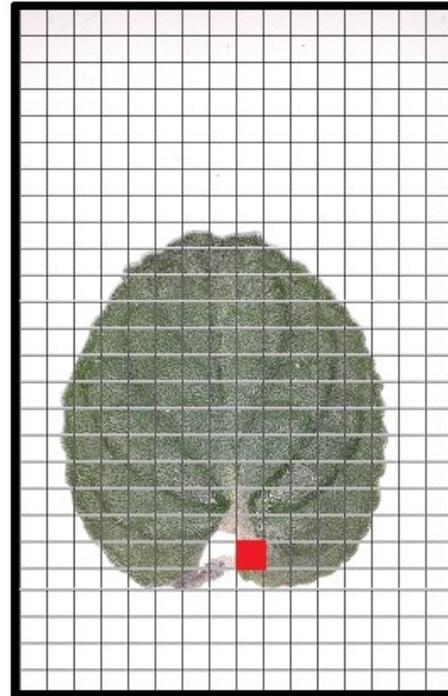
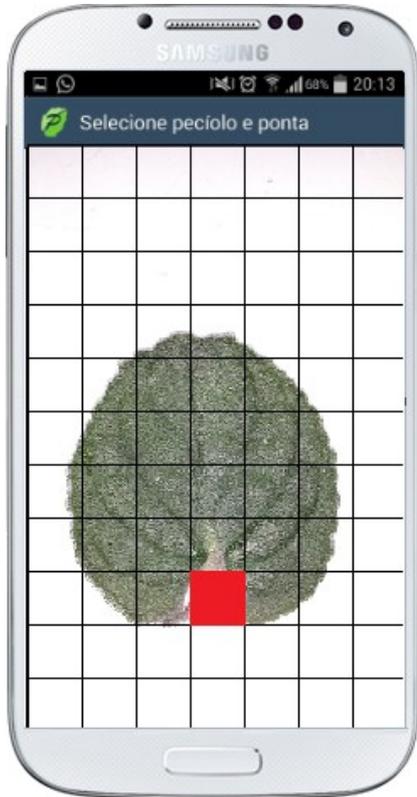
- Recebe a entrada do usuário quanto as características de textura, formato e cor
- Inicia com os valores padrão apresentados
- O checkbox desconsiderar estas características envia os valores padrão independente do que estiver selecionado

Informação de características – Seleção de pecíolo e ponta



- Primeiro o usuário deve tocar na tela indicando o pecíolo da folha, e depois a ponta
- Após selecionar as partes correspondentes o usuário pode descartar a seleção dos pontos, manter ou inverter

Informação de características – Seleção de pecíolo e ponta

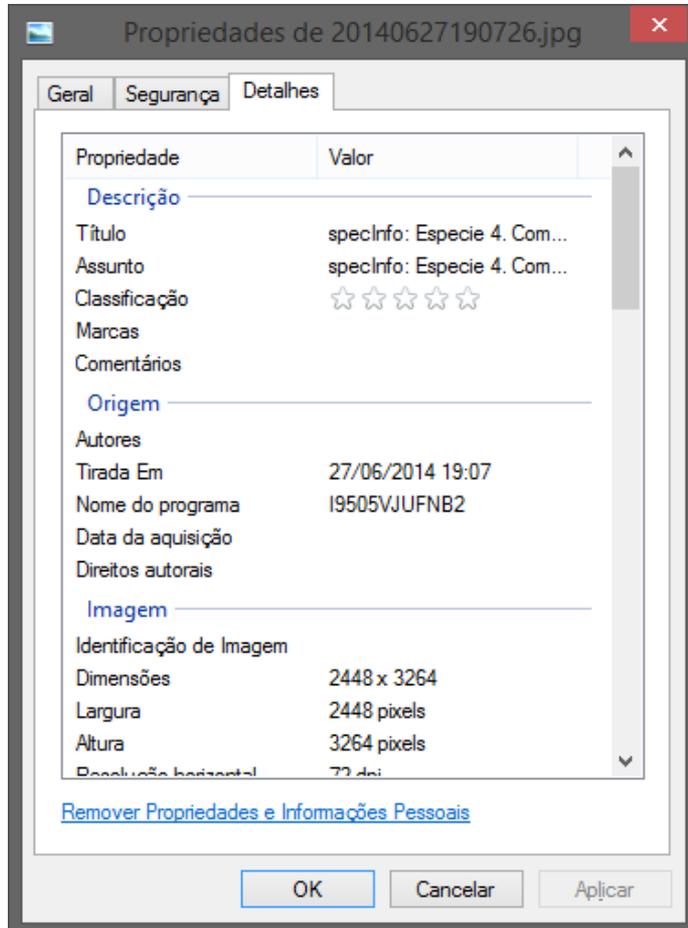


- O toque na tela retorna as coordenadas x e y da área correspondente a tela
- A imagem da câmera pode ser de resolução diferente da tela do dispositivo
- Para isso foi criado o método `getCoordinatesOfImage` que faz o cálculo para mapear corretamente o click na matriz de pixels da imagem

Envio para o servidor – obtenção da informação GPS

- Para ter acesso ao GPS do dispositivo foi criada uma classe, instanciada no momento de abertura da aplicação
- Ao inicializar, são criados `callbacks` para o evento do Android que fica atualizando periodicamente as coordenadas de localização.
- Esta classe possui um atributo do tipo `Location` que contem a localização e sempre que a localização é alterada este atributo recebe os novos valores

Envio para o servidor – Salvar as informações em EXIF na imagem

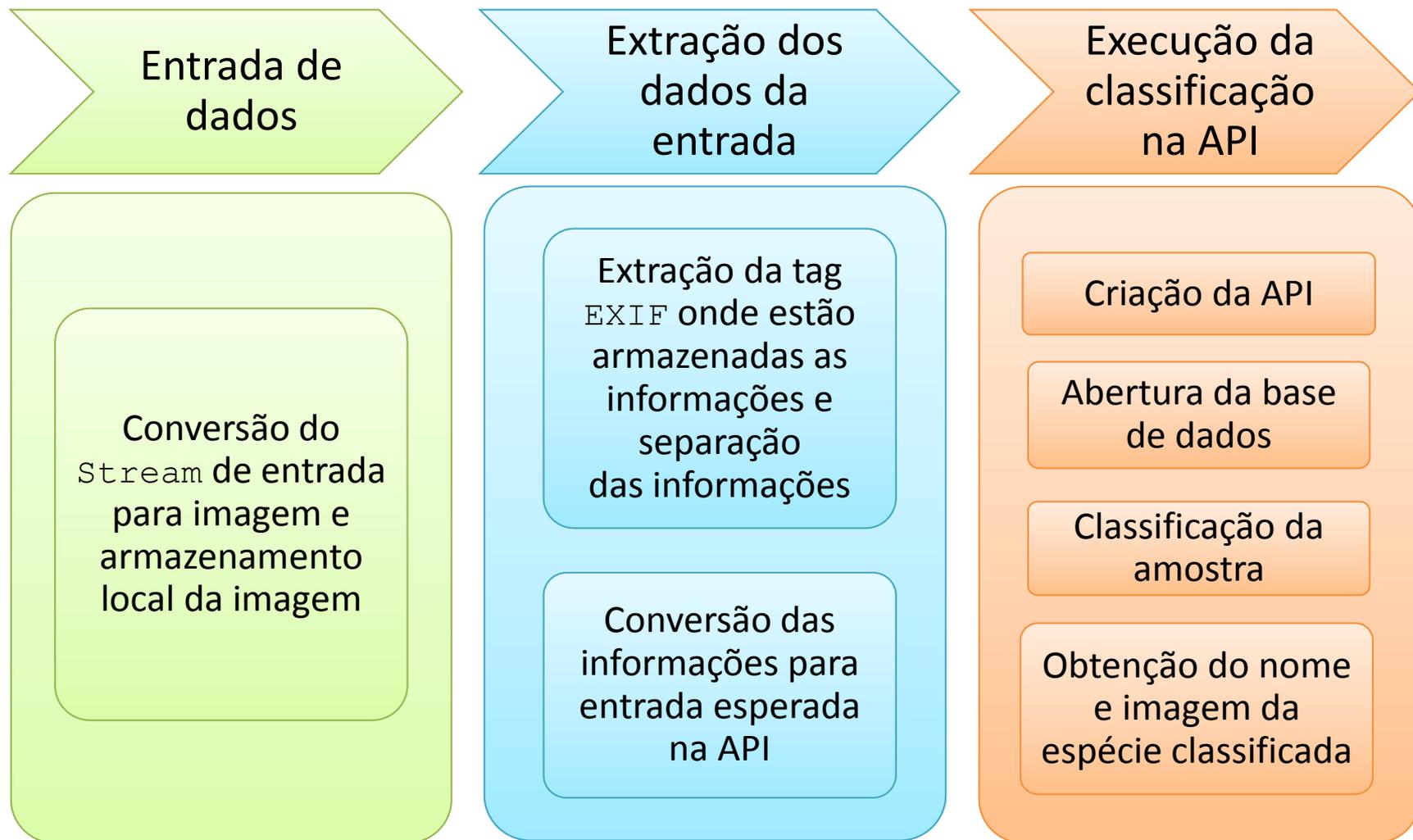


- O Android disponibiliza uma interface para trabalhar com informações EXIF, porém nenhuma tag mapeada é de texto livre
- Fez-se necessário a utilização da biblioteca `ExifDriver` de terceiros que gerenciasse estas informações
- As informações são salvas na tag `userDescription` com exceção das informações de GPS que possuem suas próprias tags

Envio para o servidor – Enviar para cadastro ou classificação

- Para trabalhar com serviços HTTP foi necessária a utilização das bibliotecas `HTTPClient`, `HTTPCore` e `HTTPMime` disponibilizadas pelo Apache
- Para o envio é necessário estar em uma thread fora da thread principal da aplicação.
- Os métodos para classificação e cadastro são do tipo POST, para utiliza-los é necessário a configuração de um corpo para a mensagem

Etapas do cadastro ou classificação na aplicação servidor



Entrada de dados - Conversão do `Stream` de entrada para imagem

- Ao atender às requisições POST, o serviço espera um `Stream` como parâmetro.
- Para a conversão para imagem, o `Stream` é lido na codificação ASCII
- São extraídos os bytes desta leitura e então é convertido para `MemoryStream` na intenção de salvá-lo.

Extração dos dados da entrada - Extração da tag EXIF

- Para a extração da tag `EXIF`, o .Net disponibiliza o método `GetPropertyItem`, que recebe como parâmetro o valor correspondente a tag na imagem
- Após extrair o texto é criado um `Dictionary` com o nome e o valor de cada característica presente no texto
- A API recebe os valores em uma `struct` por posição, então são criados os objetos `Point` para passar o pérculo e ponta, e convertido os valores das novas características por meio de um `Enum`

Testes

- Foram realizados duas etapas de teste para a validação da implementação:
 - Testes automatizados, utilizando ferramenta disponibilizada por Cassaniga, e imagens do banco de imagens Flavia *dataset*.
 - Testes manuais, utilizando somente as aplicações cliente e servidor desenvolvidas, e imagens de espécies colhidas em campo.

Testes automatizados

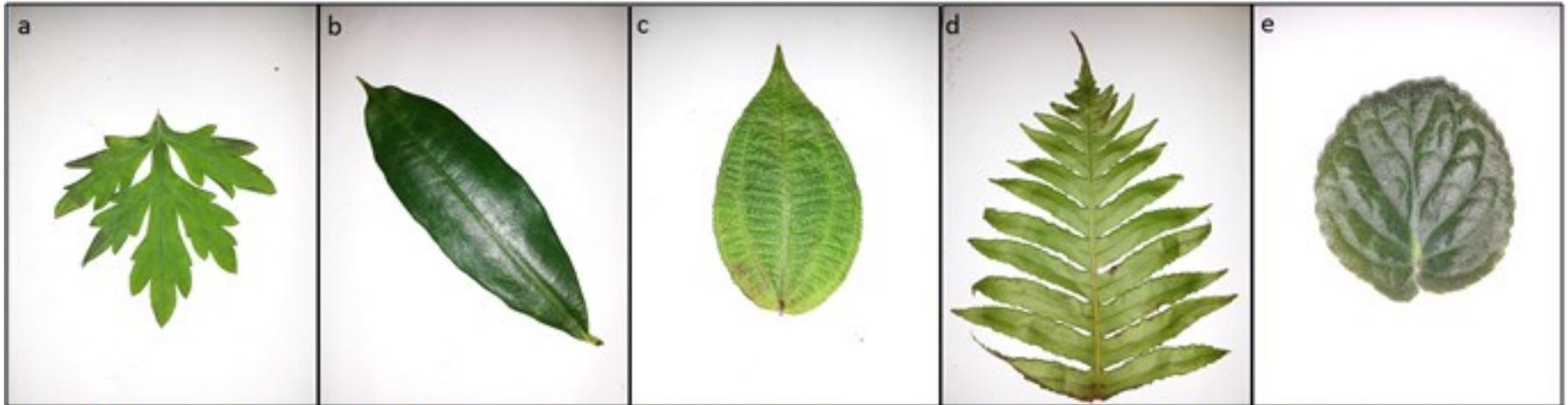
- Testes efetuados utilizando a ferramenta disponibilizada por Cassaniga. Utilizando as duas versões da API Plantarum.
- Flavia dataset, com 70% das amostras utilizadas como base de dados e o restante à serem classificadas
- Percentual de acerto era computado para cada espécie

Espécie	Numero de amostras de teste API 1.0.0	Numero de amostras reconhecidas API 1.0.0	Percentual de acerto API 1.0.0	Numero de amostras de teste API 1.1.0	Numero de amostras reconhecidas API 1.1.0	Percentual de acerto API 1.1.0
Acer Palmatum	16	16	100%	16	16	100%
Aesculus chinensis	22	22	100%	17	16	94.12%
Berberis anhweiensis Ahrendt	17	17	100%	18	18	100%
Cercis chinensis	18	18	100%	22	22	100%
Indigofera tinctoria L.	19	18	94.75%	19	19	100%
Kalopanax septemlobus (Thunb. ex A.Murr.) Koidz.	20	20	100%	17	17	100%
Phoebe nanmu (Oliv.) Gamble	17	17	100%	20	19	95%
Phyllostachys edulis (Carr.) Houz.	16	13	81.25%	16	16	100%
Total	145	141	97.24%	146	143	97.94%

Testes manuais

- Testes efetuados utilizando somente as aplicações desenvolvidas
- Foram fotografadas nove amostras de cinco espécies diferentes
- Cada espécie possuía uma combinação diferente das novas características incluídas na API

Testes: espécies utilizadas para validação da aplicação



Tipo: Composta
Pilosidade: Lisa
Discolor: Não

Tipo: Simples
Pilosidade: Lisa
Discolor: Sim

Tipo: Simples
Pilosidade: Pilosa
Discolor: Não

Tipo: Composta
Pilosidade: Lisa
Discolor: Não

Tipo: Simples
Pilosidade: Pilosa
Discolor: Sim

Testes manuais: resultados

- Foram submetidas para teste sete amostras de cinco espécies diferentes
- Foram fotografadas nove amostras de cada espécie
- Ao submeter para classificação foram informadas corretamente as novas características
- A Espécie b apresentou falso positivo com a Espécie c, ao incluir amostras com coloração mais clara.
- As demais espécies obtiveram êxito na classificação.

Tabela comparativa dos resultados

Características / Trabalhos relacionados	Sangle, Shirsat e Bhosle (2013)	Knight, Painter e Potter (2010)	Kumar et al. (2012)	Plantarum (2014)
Coordenadas de GPS	Não	Não	Sim	Sim
Extração das características	Dispositivo	Dispositivo	Servidor	Servidor
Características utilizadas	Geométricas	Geométricas	Geométricas	Geométricas, nervuras, descritores de Fourier, cor, textura, e descrição de aspecto informado pelo usuário
Método de reconhecimento	Automático	Automático	Com auxílio do usuário	Automático
Plataforma	Android	Android	iOS	Android
Precisão média	-	80	96,8	97.94

Testes manuais: resultados

- Com a mesma base de dados, onde as amostras possuíam as novas características informadas
- Ao submeter para classificação não foram informadas as novas características
- A Espécie d apresentou falso positivo com a Espécie c, ao incluir amostras com coloração mais clara
- As demais não puderam ser reconhecidas pela API

Conclusões

- Resultados satisfatórios, a aplicação permitiu o cadastro e classificação de maneira adequada utilizando as novas características implementadas
- Devido à construção de um serviço web, a construção de outras interfaces em outras plataformas é simples

Limitações

- Aplicação servidor opera apenas na plataforma Windows
- Aplicação cliente opera apenas na plataforma Android
- O processo depende de uma imagem com alto contraste entre a folha e o fundo sem ruídos

Limitações

- A aplicação cliente somente executa em dispositivos com resolução de 1920x1080
- A imagem de entrada deve ser fotografada com o dispositivo em orientação retrato
- A aplicação depende de uma base de dados previamente cadastrada para a API Plantarum
- Para o cadastro e classificação é necessário que a amostra tenha sido fotografada e salva pela própria aplicação

Extensões

- Implementação de novas características
- Permitir a utilização da câmera em qualquer orientação
- Portar a aplicação à dispositivos com diferentes tamanhos de tela
- Fazer pré-tratamento da imagem no dispositivo

Extensões

- Permitir enviar para a classificação qualquer imagem do dispositivo
- Implementar mecanismo na aplicação servidor para a criação automática da base de dados
- Implementar interface para outras plataformas de dispositivos móveis

Demonstração

Obrigado!