Andrew Zysk
12/9/2016

Design Document 1

Context Viewpoint

Project Description

The purpose of this project is to create an image processing software application for crime scene investigation. The intended user, a forensic scientist, would use the application to upload images of crime scene evidence (e.g. blood stains, debris) and automate the forensic process of bloodstain pattern analysis. The objective of the finished product is to provide a set of image analysis tools for a crime scene investigator to ascertain information about a bloodstain.

User

The intended user of the application is a crime scene investigation professional (e.g. forensic science experts, detectives) who captures or analyzes photos of crime scene evidence.

Use Cases

The following use cases are actions that can be taken by the user.

- Upload image from camera
- Open existing image
- Accept suggested actions taken
- Decline suggested actions taken
- Interact with image view without returning measurements
    - Select region of interest
    - Zoom to region of interest
    - Pan
- Interact with image view and return measurements
    - Automatically identify bloodstains
    - Show table of measurement data about identified bloodstains

- Show report on probable derived information about bloodstains, e.g. type of wound, duration of the bloodstain

Composition Viewpoint

The major components of the final product are defined as: *image interaction tools*, *executive control*, *core image analysis tools,* and *user image management*.

Image interaction tools are the generic set of tools the user applies to view the image in various ways. Image interaction tools include zooming, panning, and selecting regions of interest in an image. The image interaction tools explicitly do not include image analysis tools that output derived data from image processing algorithms. Image processing algorithms include processes such as edge detection and image segmentation.

Component	**Sub-component**	**Requirement type**	**Requirement**
Image interaction tools	Region of Interest	Functional	System must select a region of interest in the image view as input by the user
Image interaction tools	Zoom to ROI	Functional	System must zoom the image view to the region of interest as input by the user
Image interaction tools	Pan	Functional	System must pan the image view as input by the user
Image interaction tools	General	Non-functional	System shall not include image interaction tools that output image segmentation measurements, or any derived data from the image content itself

Executive control is the component that suggests actions for the user to take, based on an algorithm that identifies potential regions of interest for the user. The executive control module will suggest actions to the user through visual prompts.

Component	**Sub-component**	**Type of requirement**	**Requirement**
Executive control	Executive control prompt	Functional	System must prompt the user to take suggested actions

Core image analysis tools are the set of tools that the user applies to gather information about the bloodstains in the image. For example, bloodstain pattern analysis is a core image analysis tool that identifies bloodstain objects in the image, and outputs a table of measurements about the bloodstain segments.

Component	Sub-component	Type of requirement	Requirement
Core image analysis tools	Bloodstain pattern analysis tool	Non-functional	System must currently be viewing an image
Core image analysis tools	Bloodstain pattern analysis tool	Functional	System must set a color threshold of (R=160, G=88, B=88), and transform the image to a binary mask according to the color threshold
Core image analysis tools	Bloodstain pattern analysis tool	Functional	System must segment the image according to the binary mask, and show a resultant table with measurement data for each segment (bloodstain).
Core image analysis tools	Bloodstain pattern analysis tool	Functional	System must show a report detailing derived information about the bloodstains.
Core image analysis tools	Bloodstain pattern analysis tool	Non-functional	System shall include the following measurement data options: roundness, rectangularity, circularity, diameter, area, perimeter, center of mass
Core image analysis tools	Bloodstain pattern analysis tool	Non-functional	System shall include the following derived information attributes about the bloodstains: duration, type of wound, qualitative color of bloodstain

The **user image management** component is the component the user interfaces with to manage images, such as uploading images and managing images in the project library.

Component	**Sub-component**	**Type of requirement**	**Requirement**
User Image management	Image view	Functional	System must view a selected image in the project library upon user request
User Image management	Image view	Functional	System must view an image upon user upload of the image
User Image management	Image view	Functional	System must, upon a prompt from executive control, provide user input to either accept or decline the suggested actions from the prompt
User Image management	Image view	Non-functional	System shall be useable on mobile operating systems
User Image management	Project library	Functional	System must view a selected image in the project library upon user request
User Image management	Project library	Non-functional	System shall be accessible by the user's username and password
User Image management	Project library	Non-functional	System shall allow images from the user's accesible local directories to be uploaded
User Image management	Project library	Non-functional	System shall allow images from the user's mobile camera roll to be uploaded

Logical Viewpoint

The application follows the Model-View-Controller architectural pattern.

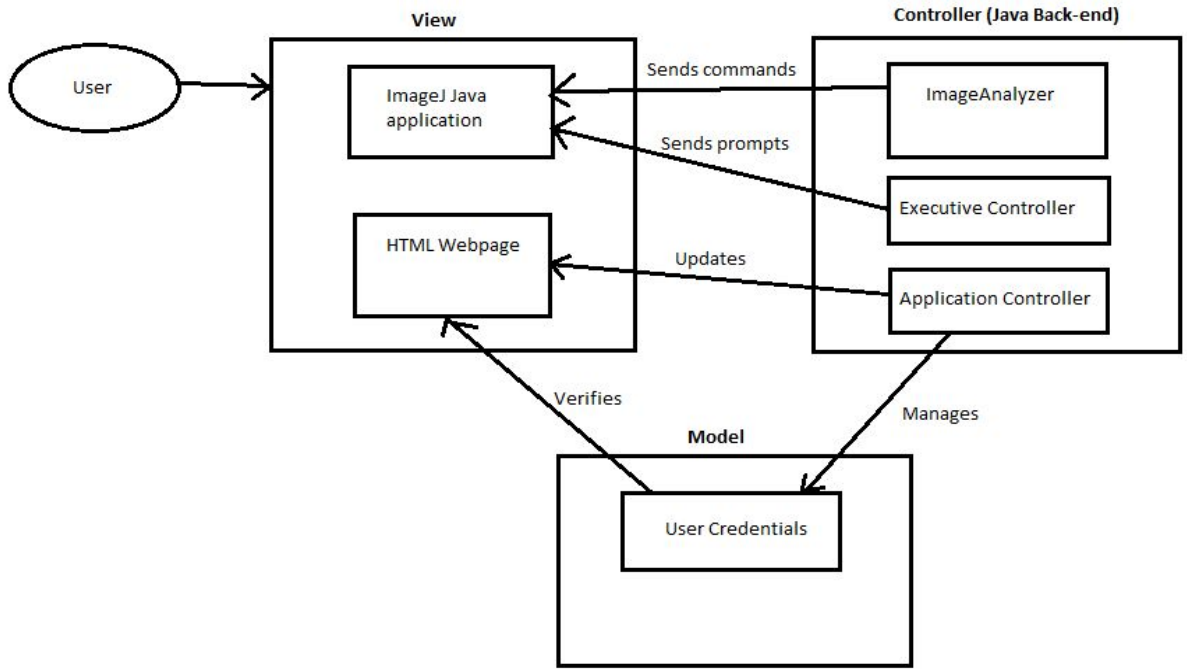

Figure 1. Project Data Flow Diagram.

The User interacts with the View, which is comprised of an HTML Webpage and the ImageJ java application. When the user logs in, the User Credentials in the Model verify whether or not the User can access his account. The User uploads and manages images on the Webpage, which listens for HTTP requests (default: port 9000) from the Application Controller, a java class. When the User views an image, the ImageJ Java application is the package that allows the Controllers to manipulate images. The ImageAnalyzer class provides the image interaction tools and the core image analysis tools, which allow the user to interact with an image. The Executive Controller class provides the executive control prompts to the ImageJ application.

User Interface

The user interface allows the user to apply image interaction tools and core image analysis tools. The set of core image analysis tools are available to the user by a toolbar. The executive control module will suggest actions for the user to take. The user may accept or decline these prompted actions, as the user has full access to the tool set whenever an image is open.

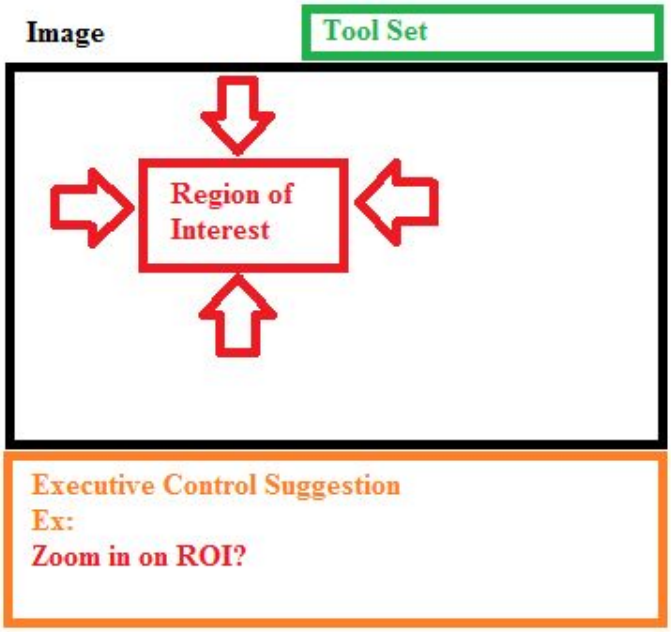

Figure 2. User Interface with Executive control suggestion

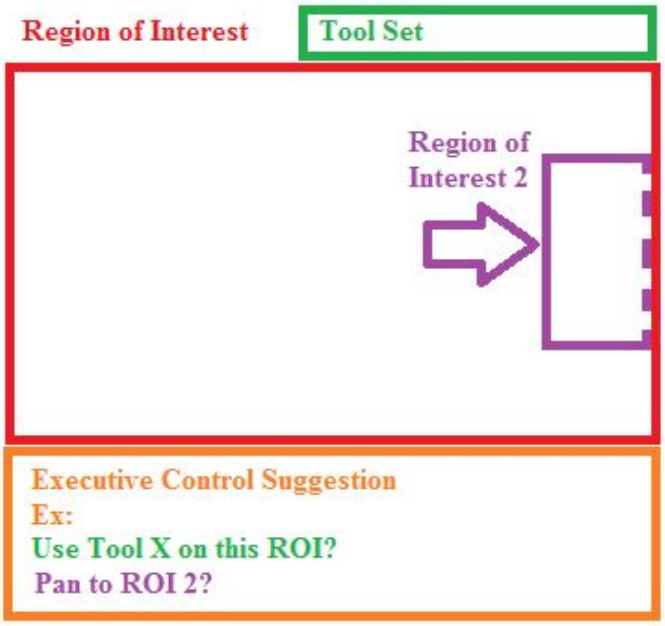

Figure 3. User Interface, accepted Executive control suggestion

Development Timeline

Controller-View Modules

Week 1-2: Application Controller and Webpage: implement basic image upload.

Week 3-6: ImageAnalyzer and ImageJ java application: implement image processing algorithms in Java. Test ImageJ functionalities such as Select and Zoom to Selection.

Week 7-8: Executive Control: develop executive control algorithms and implement using the processes determined in Week 3-6. Test with several sample images.

Week 9-11: ImageAnalyzer: Develop algorithms for core image analysis tools.

Week 12-16: View: Develop effective user interface. Consult and test with sample users.

Week 17-20: User Credentials: Implement user database and user login interface.