Software Requirements Specification for ImgBeamer: SEM Image Formation

Joachim de Fourestier

February 1, 2023

Contents

Revision History

Date	Version	Notes
2023/02/03	1.0	First version
Date 2	1.1	Notes

1 Reference Material

This section records information for easy reference.

1.1 Table of Units

Throughout this document SI (Système International d'Unités) is employed as the unit system. In addition to the basic units, several derived units are used as described below. For each unit, the symbol is given followed by a description of the unit and the SI name.

symbol	**unit**	**SI**
m	length	metre
s	time	second
Pa	pressure	pascal
J	energy	joule
K	temperature	kelvin
mol	substance	mole

Additional units used:

symbol	**unit**	**name**
px	pixel	picture element ?
nm	length	nanometre (10^{-9} m)
µm	length	micrometre (10^{-6} m)

1.2 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units. The choice of symbols was made to be consistent with the heat transfer literature and with existing documentation for solar water heating systems. The symbols are listed in alphabetical order.

symbol	**unit**	**description**
d_p	nm	probe diameter (size)
d_i	nm	probe step size in image space
d_o	nm	probe step size in object space
$I_{n\times m}$	8-bit gray level	Image intensity matrix that has n rows and m columns.

$M_{n\times m}$	boolean	Mask / stencil matrix that has n rows and m columns.
$R_{n\times m}$	8-bit gray level	Resulting image intensity matrix that has n rows and m columns.

1.3 Abbreviations and Acronyms

symbol	**description**
A	Assumption
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
TM	Theoretical Model
2D	two-dimensional
3D	three-dimensional
BSE	Backscattered Electron
CCD	Charge Coupled Device
EM	Electron Microscopy
FOV	Field of View
LM	Light Microscopy
ROI	Region of Interest
SEM	Scanning Electron Microscope
SE	Secondary Electron
ImgBeamer	SEM image formation demo tool

2 Introduction

Images formed by Scanning Electron Microscope (SEM) are created using a specific process where the image quality can be greatly influenced by the imaging parameters given by the user, aside from any inherent electron optics limitation of a given instrument. The motivation of this project is to be able visualize qualitatively the influence of the spot profile (shape and size) and the rastering parameters (such as pixel size). Since the quality of an image is very often evaluated subjectively by an experienced microscope user, the idea behind this software is provide a way to show that adjusting the imaging parameters could lead to a loss of information or misinterpretation. Is is also to prove or disprove if there is a "rule of thumb" for the optimal spot-to-pixel size ratio.

2.1 Purpose of Document

This document serves as the software specification. The details of the requirements, limitations, definitions, models to be used are laid explicitly within this document.

2.2 Scope of Requirements

The scope of the requirements of this software is restricted to only the image formation process used in SEMs, more specifically the image processing from signal to an image or visual represention. It does not cover election-sample interactions or CCD (charge coupled device) detector voltage to signal value conversion. See the assumptions section (Section 4.2.1) for further details.

2.3 Characteristics of Intended Reader

The intended reader of this document should have a basic understanding of electron-optics concepts used in an SEM. This can be equated to standard level two university physics (optics, electricity, magnetism, thermodynamics and modern physics) along with a standard undergraduate course related to materials characterization (concepts such as backscattered electrons, mean free path, and electron density). The reader should have a basic understanding of two-dimensional image processing techniques and level one university linear algebra (matrices).

2.4 Organization of Document

This document contains an introduction to the software and its goals. It is meant to be a reference document to the readers and is based on a SRS template from Smith and Lai (2005); Smith et al. (2007). The document include a description of the scope, a detailed list of the terminology, definitions, and models used, as well as a more specific and detailed description of the problem and its solutions. This document also defines the requirements, as well as the likely changes and traceability details.

3 General System Description

This section provides general information about the system. It identifies the interfaces between the system and its environment, describes the user characteristics and lists the system constraints.

[The purpose of this section is to provide general information about the system so the specific requirements in the next section will be easier to understand. The general system description section is designed to be changeable independent of changes to the functional requirements documented in the specific system description. The general system description provides a context for a family of related models. The general description can stay the same, while specific details are changed between family members. —TPLT]

3.1 System Context

[Your system context will include a figure that shows the abstract view of the software. Often in a scientific context, the program can be viewed abstractly following the design pattern of Inputs → Calculations → Outputs. The system context will therefore often follow this pattern. The user provides inputs, the system does the calculations, and then provides the outputs to the user. The figure should not show all of the inputs, just an abstract view of the main categories of inputs (like material properties, geometry, etc.). Likewise, the outputs should be presented from an abstract point of view. In some cases the diagram will show other external entities, besides the user. For instance, when the software product is a library, the user will be another software program, not an actual end user. If there are system constraints that the software must work with external libraries, these libraries can also be shown on the System Context diagram. They should only be named with a specific library name if this is required by the system constraint. —TPLT]

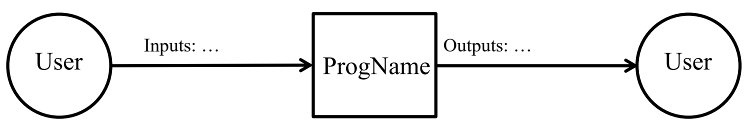


Figure 1: System Context

[For each of the entities in the system context diagram its responsibilities should be listed. Whenever possible the system should check for data quality, but for some cases the user will need to assume that responsibility. The list of responsibilites should be about the inputs and outputs only, and they should be abstract. Details should not be presented here. However, the information should not be so abstract as to just say "inputs" and "outputs". A summarizing phrase can be used to characterize the inputs. For instance, saying "material properties" provides some information, but it stays away from the detail of listing every required properties. —TPLT]

- User Responsibilities:
  -

- ImgBeamer Responsibilities:
  - Detect data type mismatch, such as a string of characters instead of a floating point number
  -

3.2 User Characteristics

The ImgBeamer software end user is anyone that has used an SEM or is learning about them. This ranges from a university undergraduate science student in which their curriculum includes electron microscopy (EM), to experienced SEM operators or even to experts in the field of EM. The software end users also includes those described in Section 2.3 (intender readers of this document). That said, image processing nor linear algebra knowledge is not required.

3.3 System Constraints

[System constraints differ from other type of requirements because they limit the developers' options in the system design and they identify how the eventual system must fit into the world. This is the only place in the SRS where design decisions can be specified. That is, the quality requirement for abstraction is relaxed here. However, system constraints should only be included if they are truly required. —TPLT]

4 Specific System Description

This section first presents the problem description, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the assumptions, theories, definitions and finally the instance models.

4.1 Problem Description

ImgBeamer is intended to help visualize and understand the influence of the spot profile and pixel size. It should be able to help determine the optimal spot-to-pixel-size ratio.

4.1.1 Terminology and Definitions

This subsection provides a list of terms that are used in the subsequent sections and their meaning, with the purpose of reducing ambiguity and making it easier to correctly understand the requirements:

Terminology	Definition
Backscattered Electron	An electron that is elastically scattered backwards to the initial direction of travel as a result of electron-sample interactions between the beam and the sample.
Beam	The continuous focused stream of electrons accelerated at high speeds down the column of an electron microscope.
Bit Depth	It is the range of the value (intensity or color) of a pixel. Specifically, it is the maximum number of bits used to represent said value.
Brightness	The intensity value (or perceived luminance for colours) of a signal, point (or pixel), or area within an image.
Contrast	The difference or distinguisable quality of an intensity (or colour) or brightness value of a point (or pixel) from another point in an image.
Electron-Sample Interactions	The interactions as a result of electrons from the beam hitting the sample surface leading to an electron collision cascade. This in produces various emissions such as x-rays, BSEs, and SEs.
Electron Microscopy	The practice of microscopes that specifically use electrons as the source of illumination to produce an image or signal.
Exact-sampling	The sampling size or rate that matches the specificity of the targeted resolution. In the context of this document, this is when the size of the probe that matches the size of the pixel.
Field of View	The observable or visible extent given by the microscope and varies with magnification.
Ground Truth	The data or information that is considered as the source or reference standard of truth.
Image	A representation of visual information that can be represented by 2D matrix where each cell represents a pixel with a specific value (representative of intensity or colour). –
Image Quality	The perceived clarity, amount of information or detail representative of reality that is contained within an image.
Image Resolution	The size dimensions (width and height) of an image. Typically, it is defined in pixels.
Interaction Volume	The volume within the sample in which there are electron-sample interactions caused by the beam coming into contact with the surface of the sample.
Light Microscopy	The practice of microscopes that specifically use light as the source of illumination to produce an image or signal.

Mean Free Path	The average distance within a system that a particle can travel before it collides with other particles.
Nyquist Limit	More formally known as the "Nyquist-Shannon" sampling limit is the minimum sampling rate at which all the essential information is captured when discretizing a continous signal, resulting in little to no distortions.
Over-sampling	This is a sampling rate or size that is too large where the specificity or origin of the information is lost. This generally results in increased confidence, but at a loss of precision.
Pixel Size	The size or dimensions of one cell within a raster grid.
Probe	Generally interchangeable with "beam". Contextually within this document, it is the final part of the beam that comes into contact with the surface of the sample (being analysed within the microscope) forming a "spot" with a targeted size and shape.
Raster Grid	A scanning pattern with a defined and repeated cell size.
Region of Interest	A defined 2D area in both position, size, and shape on a sample.
Resolution	As defined by the Rayleigh criterion, it is the minimum distance by which two points or features can be distinguished.
Secondary Electron	An electron that emitted from the sample as a result of inelastic electron-sample interactions between the beam and the sample. These can be weakly bounded valence electrons or conduction band electrons.
Signal-to-Noise Ratio	The amount of information (signal) considered useful or representative of reality versus the amount of noise.
Spot Shape	The overall shape of the spot or probe, whether is it a perfect circle or elongated ellipse, or even a halo or ring.
Spot Size	The overall diameter or average feret/caliper diameter of the probe.
Step Size	The length or distance of a "pixel" by which the probe is moved or deflected to when raster scanning the surface of a sample.
Subregion	A region or area contained within the current FOV.
Under-sampling	This is a sampling rate or size that is too small where the specificity or origin of the information is more restricted. This generally results in increased precision, but at a loss of confidence, or even accuracy.

4.1.2 Physical System Description

The physical system of ImgBeamer, as shown in Figure 2 and 3, includes the following elements:

PS1: The sample that is being imaged or studied.

PS2: An SEM which includes a BSE and/or SE detector. The electron beam interacts with the sample surface in high-vacuum producing various signals.

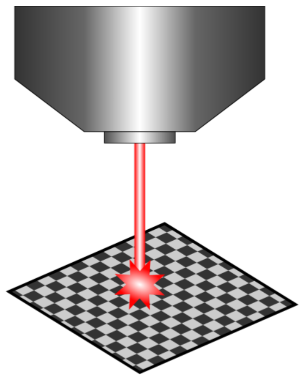


Figure 2: Schematic drawing of an electron beam hitting the sample surface within an SEM chamber. (Image from: https://github.com/joedf/ImgBeamer)

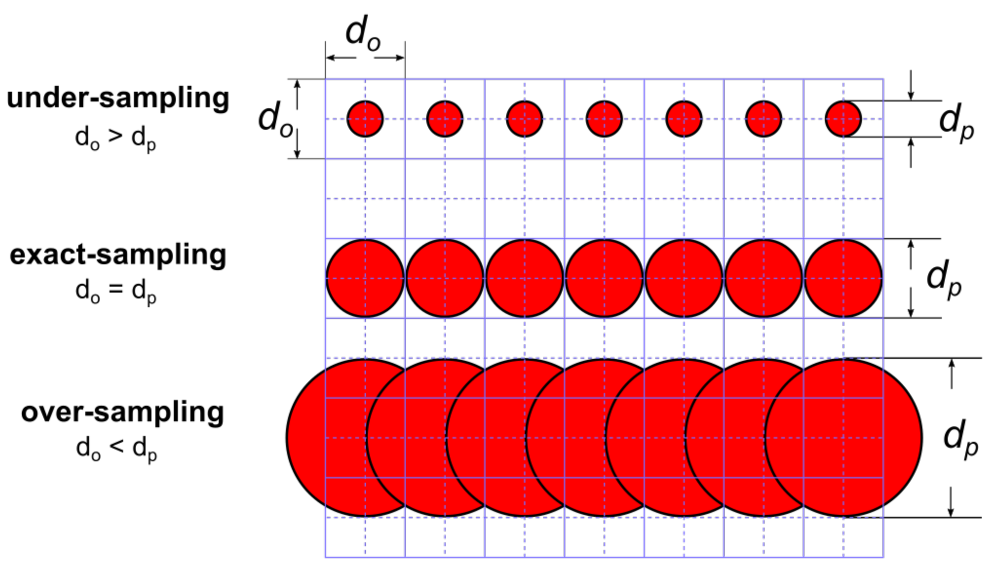


Figure 3: Schematic of the three sampling scenarios (under-sampling, exact-sampling, and over-sampling) and a visual representation of the pixel grid, d_p, and d_o. The dashed lines are to help visualize the center of each pixel cell. Adapted from Lifshin et al. (2014)

4.1.3 Goal Statements

Given an image, the goal statements are:

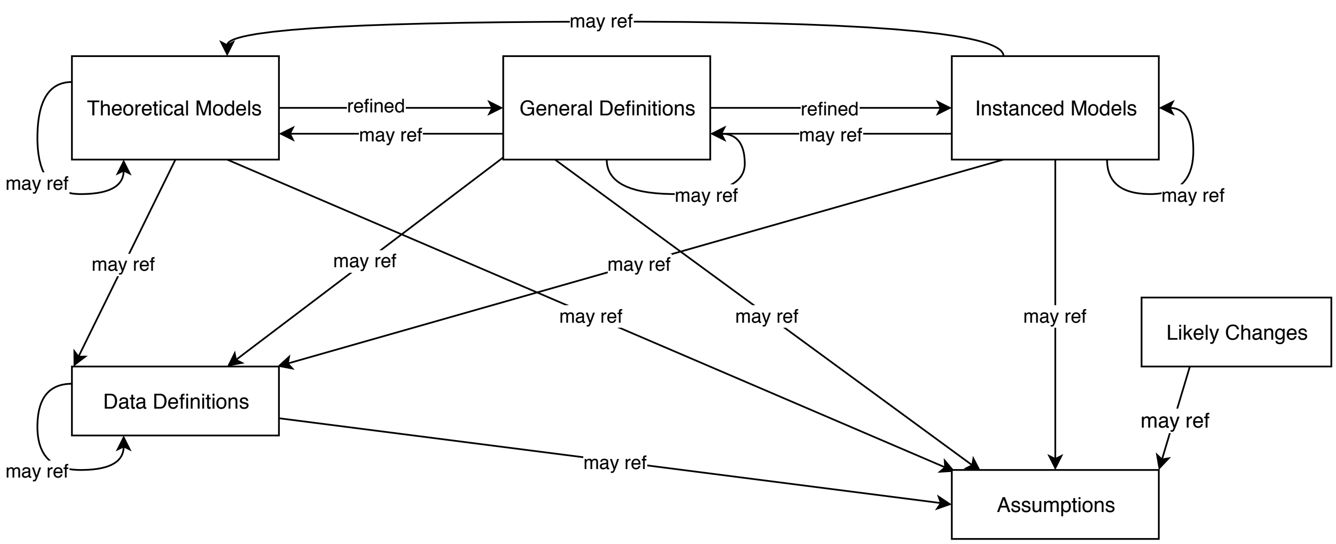
GS1: A user-friendly straight forward tool to load and reprocess an image with options following the SEM image formation process.

GS2: Provide a quantitative metric of relative image quality of the resulting image created from the given parameters and input image (as ground truth).

GS3: The ability to visualize and export the reprocessed images.

4.2 Solution Characteristics Specification

This section provides the assumptions, instance models, theoretical models, general definitions, and the data constraints. The information specfied in this section is intended to reduce ambiguities and reduce the problem into clear logical or mathematical terms.



The instance models that govern ImgBeamer are presented in Subsection 4.2.6. The information to understand the meaning of the instance models and their derivation is also presented, so that the instance models can be verified.

4.2.1 Assumptions

This section simplifies the original problem and helps in developing the theoretical model by filling in the missing information for the physical system. The numbers given in the square brackets refer to the theoretical model [TM], general definition [GD], data definition [DD], instance model [IM], or likely change [LC], in which the respective assumption is used.

A1: The sample surface is solid.

A2: The sample surface is flat (meaning no topography).

A3: The sample surface is conductive (meaning no charge build up from electrons).

A4: The sample material is made of atoms with atomic numbers greater than 3 (lithium and lighter elements are not detectable in SEM).

A5: The electron beam is ideal (meaning stable and colliminated meaning all the electrons are traveling parallel to each other down towards the sample surface within high vacuum for optimal mean free path).

A6: The beam intensity or current is considered uniform throughout the beam.

A7: The beam is orthogonal to the sample surface.

A8: The input image of the sample is considered to have infinite resolution representive of reality: ground truth.

A9: The signal yields (such as BSEs and SEs) are high enough to produce meaning optimal SNR (enough signal and noise is low).

A10: The SEM is within a controlled environment: climate controlled (temperature and humidity) and low vibration room.

A11: The image produced by the SEM as the electron raster scan the sample surface is an approximate representation of reality.

4.2.2 Theoretical Models

This section focuses on the general equations and laws that ImgBeamer is based on.

Number	TM1
Label	**Mean Free Path**
Equation	$\lambda = \frac{RT}{\sqrt{2}\pi d^2 N_A P}$
Description	λ is the mean free path (m) R is the universal gas constant = 8.3145 J/mol K T is the temperature (K) d is the molecular diameter (m) P is the pressure (Pa) N_A is Avogadro's number = 6.0221×10^{23} mol^{-1}
Notes	–
Sources	http://hyperphysics.phy-astr.gsu.edu/hbase/Kinetic/menfre.html
Ref. By	A5

Number	TM2
Label	**Contrast**
Equation	$C_{tr} = (S_{max} - S_{min})/S_{max}$
Description	S is the emitted signal strength or intensity
Notes	$S_{max} > S_{min}$
Sources	Goldstein et al. (2018), Lifshin et al. (2014)
Ref. By	–

Number	TM3
Label	**Magnification**
Equation	$M = \frac{d_i}{d_o}$
Description	M is the magnification ratio (unitless) d_i is the probe step size in image space d_o is the probe step size in object space
Notes	–
Sources	Lifshin et al. (2014)
Ref. By	–

4.2.3 General Definitions

[General Definitions (GDs) are a refinement of one or more TMs, and/or of other GDs. The GDs are less abstract than the TMs. Generally the reduction in abstraction is possible through invoking (using/referencing) Assumptions. For instance, the TM could be Newton's Law of Cooling stated abstracting. The GD could take the general law and apply it to get a 1D equation. —TPLT]

This section collects the laws and equations that will be used in building the instance models.

[Some projects may not have any content for this section, but the section heading should be kept. —TPLT] [Modify the examples below for your problem, and add additional definitions as appropriate. —TPLT]

Number	GD1
Label	**Scan or Raster Grid / Deflection Space**
Units	nm, µm, or px (pixels)
Equation	$G(x, y)$ where x and y range from 0 to inf.
Description	Scan grid...?
Source	Citation here
Ref. By	–

Detailed derivation of simplified rate of change of temperature

[This may be necessary when the necessary information does not fit in the description field. —TPLT] [Derivations are important for justifying a given GD. You want it to be clear where the equation came from. —TPLT]

4.2.4 Data Definitions

[The Data Definitions are definitions of symbols and equations that are given for the problem. They are not derived; they are simply used by other models. For instance, if a problem depends on density, there may be a data definition for the equation defining density. The DDs are given information that you can use in your other modules. —TPLT]

[All Data Definitions should be used (referenced) by at least one other model. —TPLT]

This section collects and defines all the data needed to build the instance models. The dimension of each quantity is also given. [Modify the examples below for your problem, and add additional definitions as appropriate. —TPLT]

Number	DD1
Label	**Signal yield / intensity**
Symbol	η (for BSE), δ (for SE), Y (for general)
Units	ratio - unitless
Equation	$Y_{signal} = N_{signal}/N_B$
Description	Y_{signal} is the intensity relative to the incident beam. N_{signal} is the number of scattered or emitted electrons. N_B is the number of (primary) incident electrons from the beam.
Notes	The *signal* can BSE or SE.
Sources	Goldstein et al. (2018)
Ref. By	–

4.2.5 Data Types

[This section is optional. In many scientific computing programs it isn't necessary, since the inputs and outpus are straightforward types, like reals, integers, and sequences of reals and integers. However, for some problems it is very helpful to capture the type information. —TPLT]

[The data types are not derived; they are simply stated and used by other models. —TPLT]

[All data types must be used by at least one of the models. —TPLT]

[For the mathematical notation for expressing types, the recommendation is to use the notation of Hoffman and Strooper (1995). —TPLT]

This section collects and defines all the data types needed to document the models. [Modify the examples below for your problem, and add additional definitions as appropriate. —TPLT]

Type Name	Name for Type
Type Def	mathematical definition of the type
Description	description here
Sources	Citation here, if the type is borrowed from another source

4.2.6 Instance Models

This section transforms the problem defined in Section 4.1 into one which is expressed in mathematical terms. It uses concrete symbols defined in Section 4.2.4 to replace the abstract symbols in the models identified in Sections 4.2.2 and 4.2.3.

The goal of reprocessing an image GS1 will be solved by IM1. The goal of providing an image quality metric will be solved by IM2.

Number	IM1
Label	**Bit blit (as known as Bit Block Transfer)**
Input	an image $I_{n \times m}$ and a mask $M_{n \times m}$
Output	reprocessed image $R_{n \times m}$
Description	Given an image and mask (or stencil), a boolean operation is performed iteratively on each cell of the image matrix, based on the corresponding cell value found in the mask matrix at row i and column j. If the mask cell value (at i, j) is true (or any non-zero value), the corresponding image cell value is transfered/kept in the resulting/destination matrix. Otherwise, if the corresponding matrix cell (at i, j) is false (zero), then no value is transfered, and zero will inserted at i, j in the destination image matrix.
Sources	https://en.wikipedia.org/wiki/Bit_blit
Ref. By	R6

Number	IM2
Label	**Image quality metric / evaluation**
Input	a image $I_{n\times m}$ serving as ground truth and the image $R_{n\times m}$ to compare
Output	a positive ratio value ranging from 0 to 1.00.
Description	Given the original image (ground truth) and a reprocessed image, a metric must be calculated to express image similarity where 1.00 means a perfect match and 0.00 means absolutely no correlation or similarity found. If the images do no match in size, the smaller (or larger?) image shall be scaled to match. The values do not need to be absolutely quantitative. They only need to be relatively comparable when only changing d_p or the spot shape.
Sources	
Ref. By	R7, NFR2

4.2.7 Input Data Constraints

Table 2 shows the data constraints on the input output variables. The column for physical constraints gives the physical limitations on the range of values that can be taken by the variable. The column for software constraints restricts the range of inputs to reasonable values. The software constraints will be helpful in the design stage for picking suitable algorithms. The constraints are conservative, to give the user of the model the flexibility to experiment with unusual situations. The column of typical values is intended to provide a feel for a common scenario. The uncertainty column provides an estimate of the confidence with which the physical quantities can be measured. This information would be part of the input if one were performing an uncertainty quantification exercise.

The specification parameters in Table 2 are listed in Table 3.

Table 2: Input Variables

Var	Physical Constraints	Software Constraints	Typical Value	Uncertainty
L	$L > 0$	$L_{\text{min}} \leq L \leq L_{\text{max}}$	1.5 m	10%

(*) [you might need to add some notes or clarifications —TPLT]

Table 3: Specification Parameter Values

Var	Value
L_{min}	0.1 m

4.2.8 Properties of a Correct Solution

A correct solution must exhibit [fill in the details —TPLT]. [These properties are in addition to the stated requirements. There is no need to repeat the requirements here. These additional properties may not exist for every problem. Examples include conservation laws (like conservation of energy or mass) and known constraints on outputs, which are usually summarized in tabular form. A sample table is shown in Table 4 —TPLT]

Table 4: Output Variables

Var	Physical Constraints
T_W	$T_{\text{init}} \leq T_W \leq T_C$ (by A**??**)

[This section is not for test cases or techniques for verification and validation. Those topics will be addressed in the Verification and Validation plan. —TPLT]

5 Requirements

This section provides the functional requirements, the business tasks that the software is expected to complete, and the nonfunctional requirements, the qualities that the software is expected to exhibit.

5.1 Functional Requirements

R1: Accept an input image in the following formats:

- PNG (Portable Net Graphics)
- JPG (Joint Photographic Experts Group)
- BMP (Bitmap)
- preloaded example images

R2: Accept user input of the size and shape of the spot profile and ensure that all its defining values are positive real numbers.

R3: Accept user input of pixel size and ensure it is a positive real number.

R4: Accept user input to specify a subregion (FOV) for processing.

R5: Display a representative spot layout used for processing the image.

R6: Reprocess (using IM1) and display the resulting image to the user.

R7: Calculate and display an image quality metric (according to IM2) to the user.

5.2 Nonfunctional Requirements

The key nonfunctional requirements of this software are accuracy, usability, maintainability, and portability. These are listed below in detail:

NFR1: **Accuracy** The images produced should be not manipulated to produce results that subjectively represent the opinions of the author(s) or developer(s). Thus, the software should transparently follow the specifications and be verifiable by an expert in field. The trends in image quality metric are more important the metric values themselves.

NFR2: **Usability** The user should be able to intuitively use the software and quickly understand what is being displayed. There should be little to no setup required. Although the intended use is more of a qualitative nature, the user should be able to grasp any trends in the change of the image quality metric when the input parameters are changed. The software should have a simple user interface and be responsive when given a relatively small input image (ref to assumptions here?).

NFR3: **Maintainability** The code should follow a consistent style and be reasonably separated in multiple files were it makes sense. Function names should be no longer than 40 characters. Duplicate code should be avoided wherever possible. Comments should be plentiful, but short, and avoid any unnecessary use jargon or domain-specific terms.

NFR4: **Portability** The software should be cross-platorm (Windows, Linux, MacOS) with little to no setup required. This can be in the form of a web-application in an HTML5 compliant web-browser.

6 Likely Changes

Here are changes listed that will likely be implemented as the software is improved.

LC1: Unknown for now...

LC2: [Give the likely changes, with a reference to the related assumption (aref), as appropriate. —TPLT]

7 Traceability Matrices and Graphs

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Every time a component is changed, the items in the column of that component that are marked with an "X" may have to be modified as well. Table 5 shows the dependencies of theoretical models, general definitions, data definitions, and instance models with each other. Table 6 shows the dependencies of instance models, requirements, and data constraints on each other. Table 7 shows the dependencies of theoretical models, general definitions, data definitions, instance models, and likely changes on the assumptions.

[You will have to modify these tables for your problem. —TPLT]

[The traceability matrix is not generally symmetric. If GD1 uses A1, that means that GD1's derivation or presentation requires invocation of A1. A1 does not use GD1. A1 is "used by" GD1. —TPLT]

[The traceability matrix is challenging to maintain manually. Please do your best. In the future tools (like Drasil) will make this much easier. —TPLT]

The purpose of the traceability graphs is also to provide easy references on what has to be additionally modified if a certain component is changed. The arrows in the graphs represent dependencies. The component at the tail of an arrow is depended on by the component at the head of that arrow. Therefore, if a component is changed, the components that it points to should also be changed. Figure **??** shows the dependencies of theoretical models, general definitions, data definitions, instance models, likely changes, and assumptions on each other. Figure **??** shows the dependencies of instance models, requirements, and data constraints on each other.

	T??	T??	T??	GD??	GD??	DD??	DD??	DD??	DD??	IM??	IM??	IM??	IM
T??													
T??			X										
T??													
GD??													
GD??	X												
DD??				X									
DD??				X									
DD??													
DD??								X					
IM??					X	X	X				X		
IM??					X		X		X	X			X
IM??		X											
IM??		X	X				X	X	X		X		

Table 5: Traceability Matrix Showing the Connections Between Items of Different Sections

	IM??	IM??	IM??	IM??	4.2.7	R??	R??
IM??		X				X	X
IM??	X			X		X	X
IM??						X	X
IM??		X				X	X
R??							
R??						X	
R??					X		
R??	X	X				X	X
R??	X						
R??		X					
R??			X				
R??				X			
R??			X	X			
R??		X					
R??		X					

Table 6: Traceability Matrix Showing the Connections Between Requirements and Instance Models

	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??
T??	X																		
T??																			
T??																			
GD??		X																	
GD??			X	X	X	X													
DD??							X	X	X										
DD??			X	X						X									
DD??																			
DD??																			
IM??											X	X		X	X	X			X
IM??												X	X			X	X	X	
IM??														X					X
IM??													X					X	
LC??				X															
LC??								X											
LC??									X										
LC??											X								
LC??												X							
LC??															X				

Table 7: Traceability Matrix Showing the Connections Between Assumptions and Other Items

8 Development Plan

[This section is optional. It is used to explain the plan for developing the software. In particular, this section gives a list of the order in which the requirements will be implemented. In the context of a course this is where you can indicate which requirements will be implemented as part of the course, and which will be "faked" as future work. This section can be organized as a prioritized list of requirements, or it could should the requirements that will be implemented for "phase 1", "phase 2", etc. —TPLT]

9 Values of Auxiliary Constants

[Show the values of the symbolic parameters introduced in the report. —TPLT]

[The definition of the requirements will likely call for SYMBOLIC_CONSTANTS. Their values are defined in this section for easy maintenance. —TPLT]

[The value of FRACTION, for the Maintainability NFR would be given here. —TPLT]

References

Joseph I. Goldstein, Dale E. Newbury, Joseph R. Michael, Nicholas W. M. Ritchie, John Henry J. Scott, and David C. Joy. Image Formation. In *Scanning Electron Microscopy and X-Ray Microanalysis*, pages 93–110. Springer, New York, NY, 2018. ISBN 978-1-4939-6676-9. doi: 10.1007/978-1-4939-6676-9_6. URL https://doi.org/10.1007/978-1-4939-6676-9_6.

Daniel M. Hoffman and Paul A. Strooper. *Software Design, Automated Testing, and Maintenance: A Practical Approach.* International Thomson Computer Press, New York, NY, USA, 1995. URL http://citeseer.ist.psu.edu/428727.html.

Eric Lifshin, Yudhishthir P. Kandel, and Richard L. Moore. Improving Scanning Electron Microscope Resolution for Near Planar Samples Through the Use of Image Restoration. *Microscopy and Microanalysis*, 20(1):78–89, February 2014. ISSN 1431-9276, 1435-8115. doi: 10.1017/S1431927613013688. URL https://www.cambridge.org/core/journals/microscopy-and-microanalysis/article/improving-scanning-electron-microscope-resolution-for-near-planar-samples-through-the-use-of-image-r A8715B3B599E6178E189B1887803FC79#. Publisher: Cambridge University Press.

W. Spencer Smith and Lei Lai. A new requirements template for scientific computing. In J. Ralyté, P. Ȧgerfalk, and N. Kraiem, editors, *Proceedings of the First International Workshop on Situational Requirements Engineering Processes – Methods, Techniques and Tools to Support Situation-Specific Requirements Engineering Processes, SREP'05*, pages 107–121, Paris, France, 2005. In conjunction with 13th IEEE International Requirements Engineering Conference.

W. Spencer Smith, Lei Lai, and Ridha Khedri. Requirements analysis for engineering computation: A systematic approach for improving software reliability. *Reliable Computing, Special Issue on Reliable Engineering Computation*, 13(1):83–107, February 2007.