



ИНСТИТУТ ЗА МАТЕМАТИКУ И ИНФОРМАТИКУ
ПРИРОДНО-МАТЕМАТИЧКИ ФАКУЛТЕТ
УНИВЕРЗИТЕТ У КРАГУЈЕВЦУ

МАСТЕР РАД

Примена алгоритама препознавања
облика у неутронској дозиметрији

Студент:
Никола Николић

Професор:
др Милош Ивановић

Септембар 2014.

Садржај

1	Увод у неутронску дозиметрију	3
1.1	Атоми и радијација	3
1.1.1	Типови радијације	4
1.1.2	Извори, примена и ефекти радијације	5
1.2	Неутронска радијација	6
1.2.1	Примена неутронске радијације	6
1.2.2	Дозиметрија	7
1.3	Дефиниција проблема	8
2	Обрада слика употребом <i>OpenCV</i> библиотеке	9
2.1	Сиво скалирана слика	10
2.2	Сегментација слике	11
2.2.1	Раздвајање трагова од позадине	12
2.2.2	Проналажење трагова	13
2.2.3	Сегментација трагова	15
2.3	Исправљање ротације слике у односу на претходну слику из секвенце	15
2.3.1	Детекција интересних тачака	17
2.3.2	Рачунање описног вектора	17
2.3.3	Поређење описних вектора и одређивање ротације слике	17
3	Развој корисничког интерфејса употребом <i>Qt</i> фрејмворка	19
3.1	Развојни алати	20
3.2	Модули	20
3.3	Сигнали и слотови	21
4	Софтвер за детекцију трагова неутронске дозиметрије	23
4.1	Структура пројекта	23
4.2	Начин употребе	23
4.2.1	Управљање пројектима	23
4.2.2	Управљање прегледом слике	23
4.2.3	Приказ и извоз резултата обраде	24
4.2.4	Корекција обраде	24
5	Закључак	25

Сажетак

Дозиметрија се бави одређивањем апсорбоване дозе јонизујућег зрачења у разним материјалима, укључујући и људско ткиво. Карактеристике тренутно доступних личних неутронских дозиметара су далеко од идеалних, а траг детектори се данас сматрају једном од најперспективнијих техника. Међу њима је најпознатији *CR-39* детектор. При интеракцији упадних неутрона са детектором настају трагови који нагризањем материјала базом постају видљиви. На основу тих трагова врши се процена дозе зрачења.

Мерење густине трагова врши се визуелном проценом броја трагова по јединици површине помоћу микроскопа. Тај процес може да траје и по више сати. У овом раду ће се описати примена алгоритама обраде слика (*image processing*) која овај процес може делимичнио аутоматизовати и значајно убрзати.

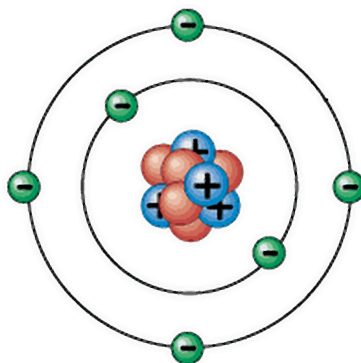
У првом поглављу дат је увод у неутронску дозиметрију, њену примену и дефинисан је проблем који ће решавати овај рад. Друго поглавље представља *OpenCV* библиотеку и описује коришћене алгоритме за обраду слика. Треће поглавље укратко представља *Qt* фрејмворк за развој корисничког интерфејста и његове главне предности. У четвртом поглављу описан је софтвер за детекцију трагова неутронске дозиметрије. Дат је опис структуре пројекта као и начин употребе софтвера. У последњем, петом поглављу дата је анализа софтвера, као и предлог могућих унапређења.

Глава 1

Увод у неутронску дозимертију

1.1 Атоми и радијација

Атом је основна јединица материје. Атоми су изграђени од позитивно наелектрисаног атомског језгра и одређеног броја електрона. Електрони носе негативно наелектрисање и крећу се у електронском омотачу око језгра. Атомско језгро садржи протоне, који имају позитивно наелектрисање, једнако негативном наелектрисању електрона и неутроне који немају наелектрисање.



Слика 1.1: Шематски приказ структуре атома

Хемијске особине атома одређује број протона у језгру. Сваки атом садржи једнак број протона и електрона и зато је електрично неутралан. Атом постаје наелектрисан тако што прими или отпусти један или више електрона и постаје јон. Такав процес назива се јонизација.

Како хемијске особине атома не зависе од броја неутрона, постоје атоми истог елемента са различитим бројем неутрона - изотопи. Атомска језгра са неповољним односом броја протона и броја неутрона су нестабилна и путем радиоактивног распада прелазе у стабилније стање.

Сама реч радијација потиче од латинске речи *radiare* која у преводу значи испуштати зраке. Радиоактивност је спонтани процес у којем се атомско језгро, емитујући једну или више честица преображава у друго језгро. При трансформацији језгра, мења се састав или енергетско стање.

Радиоактивно зрачење продире кроз различите материјале, а такође може и да јонизује средину кроз коју пролази. На тај начин може да изазове оштећење ћелија

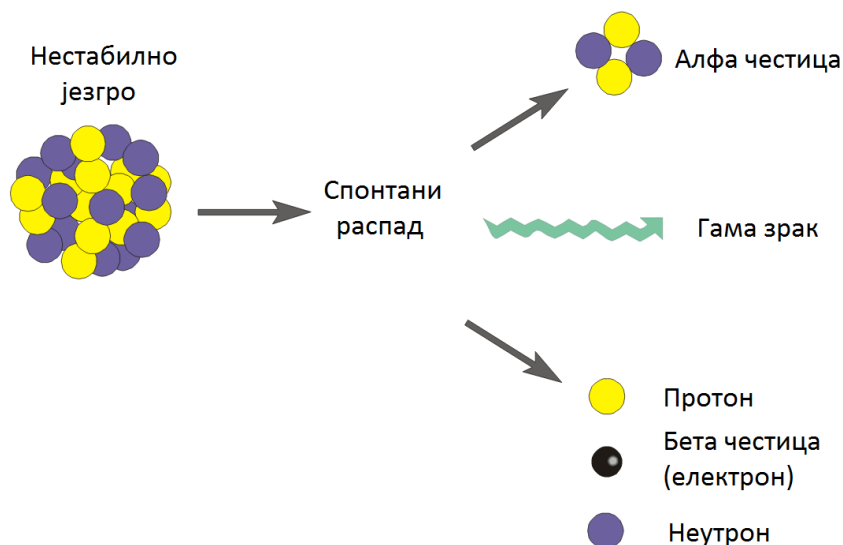
живих организама. Настали јони нарушавају биохемијске процесе у ћелијама, што може довести до разних поремећаја у њиховом функционисању и дељењу, па и до настанка озбиљних болести.

1.1.1 Типови радијације

Према ефектима које производи на материју, радијација се може класификовати на јонизирајуће и нејонизирајуће зрачење.

Јонизирајуће зрачење укључује космичке зраке, X зраке и радијацију од радиоактивних материјала. Заједничко својство је велика енергија која омогућава јонизацију. Како долази до јонизације атома и молекула који су до тада били неутрални, настају оштећења материјала. Јонизирајуће зрачење спада у најопасније, оно је за око невидљиво, не осећа се код контаминације, тешко се детектује и постоји јако узак број терапија које могу помоћи. Три основна типа јонизирајућег зрачења су α , β и γ зрачење.

Нејонизирајуће зрачење укључује ултраљубичасто светло, топлотно зрачење, радио таласе и слично. Не поседује енергију као јонизирајуће зрачење и обично се сматра безопасним при малим снагама које не изазивају пораст температуре.



Слика 1.2: Могуће честице при распаду атомског језгра

- **Алфа радијација (α)** је позитивно наелектрисано језгро хелијума (језгро са два протона и два неутрона) емитовано од већ нестабилног језгра. Алфа честица је релативно масивна, али има кратак домет у ваздуху, свега 3-5 *cm*. Алфа честице могу апсорбовати кожа или папир.
- **Бета радијација (β)** Бета честице су вискоенергетски електрони или позитрони. Негативни електрони настају распадом језгра са вишком неутрона, а позитивни електрони настају при распаду језгра са вишком протона од оног потребног за стабилност језгра. Бета честице су много мање од α честица и могу продрети

даље у материјал или ткиво. Бета радијација се може апсорбовати материјалима од пластике, стакла или метала.

- **Гама радијација (γ)** је врло високи енергетски фотон (форма електромагнетне радијације врло кратке дебљине таласа) емитован од нестабилног језгра и често у исто време емитује и α и β честице. Гама радијација узрокује јонизацију у атомима када пролази кроз материјал, првенствено због интеракције са електронима. У одсуству густих материјала, γ зрачење може оштетити материјал на који делује.
- **Неутронска радијација (n)** је емитовање неутрона из нестабилног језгра. Ван језгра неутрони су нестабилни и имају време полураспада око 11.5 минута. Када су у реакцији са материјалом изазивају емисију β и γ радијације. Неутронска радијација захтева велику заштиту да би редуковала излагање.

1.1.2 Извори, примена и ефекти радијације

Изворе радијације делимо на **природне** и **вештачке**. Природној радијацији изложени смо кроз зрачење Сунца и космоса, радиоактивних ствари у тлу, кућама којима живимо, преко хране и пића које уносимо. Количина овог неизбежног природног зрачења се разликује од места до места на Земљи.

Ефекти радијације на људско здравље од природних извора нису изразито негативни јер се оно никад не сакупља у телу. Природно зрачење је стално и веома слабо. Биолошки механизам је прилагођен еволутивно на то зрачење.

Вештачки извори радијације на земљи премашују 5% укупне радијације. Ове изворе радијације су они које је човек изградио:

- **Нуклеарне електране** имају за циљ да произведу електричну енергију. Највећу и сталну потенцијалну опасност у раду нуклеарних електрана чине евентуални инциденти при којима се емитују велике количине радиоактивног материјала.
- **Медицински извори** радијације настају при лечењу неких болести услед употребе апарата који емитују радиоактивно зрачење. Такође, истрошени апарати и опрема остају као проблем у виду радиоактивног материјала.
- **Технички извори** радијације се односе на разне апарате који функционишу на бази радионуклеида те представљају опасност за животну средину (радиоактивни громобрани, разни апарати у научно-истраживачким институцијама).

Јонизирајуће зрачење делује штетно на биолошке системе и може да доведе до појаве функционалних, морфолошких и генетских промена, а уколико су дозе којима је особа изложена високе може да дође и до смрти. Нису сви људи подједнако осетљиви на зрачење, као ни сва ткива и органи у организму.

Акутни ефекти се јављају непосредно после озрачивања особе код изложености високим дозама радијације. Јачина ефекта зависи од примљене дозе, врсте зрачења и осетљивости озрачених ткива. Код акутне озрачености јавља се радијациона болест.

Код хроничне изложености радиоактивном зрачењу, обично код професионално изложених особа, долази до оштећења ткива и то најчешће крви и хематопоетских органа. Уколико се реагује на време, најчешће долази до опоравка.

1.2 Неутронска радијација

Неутрони су ненаелектрисани и не реагују са електронима, па могу да прођу знатно растојање у материји без интеракције. Понашање неутрона је слично γ зрачењу, и оно припада групи индиректно јонизирајућих зрачења. Неутрони не могу директно да јонизују материју, већ то чине искључиво преко секундарно произведених честица.

Електромагнетска интеракција неутрона са електронима је занемарљива. Пролазећи кроз материју неутрони се могу расејати еластично или нееластично на језгрима материје. Расејање је еластично ако се укупна кинетичка енергија очува у судару тј. када је енергија коју неутрон изгуби једнака кинетичкој енергији узмака језгра. Расејање је нееластично када језгро апсорбује извесну енергију и пређе у више енергетско стање. Такође, неутрон може бити захваћен или апсорбован од стране језгра што доводи до промене броја протона или неутрона захваћеног језгра, па самим тим и његових физичких и хемијских својстава.

За брзе неутроне је типично да највише губе енергију у материји преко еластичних расејања. Овај процес се назива модерација неутрона и један је од најважнијих процеса у нуклеарним реакторима. Како енергија неутрона опада расејање се наставља, али генерално, расте вероватноћа захвата на језгру. Дифузија термалних неутрона је такође узрокована низом еластичних расејања.

Еластично расејање неутрона се доста користи за регистровање брзих неутрона проучавајући узмакнута језгра (углавном узмакнуте протоне) различитим инструментима. Овај тип расејања се такође користи за регистровање узмакнутих језгара помоћу јонизационих метода.

1.2.1 Примена неутронске радијације

Најважније примене неутронских извора су у индустрији, у науци и истраживању, као и у медицини.

- **Геофизичка мерења:** мапирање и анализа рудника минерала, нафтна истраживања, мапирање и анализа каменолома, истраживање урана.
- **Контрола индустријских процеса:** контрола цементног процеса, испитивање квалитета угља, испитивање дебљине зида.
- **Медицина:** мерење састава тела, терапија канцера, студије дијете и исхране.
- **Безбедност:** детекција и идентификација експлозива, детекција и идентификација хемијског оружја, детекција и идентификација специјалних нуклеарних материјала, детекција нагазних мина.
- **Општа истраживања:** референтни извор брзих неутрона за инструментацију, калибрациони извор за инструментацију за посматрање неутрина, проучавање осетљивости електронских компоненти на радијацију, испитивање нуклеарног реактора, неутронска радиографија.
- **Екологија:** испитивање нуклеарног отпада, испитивање отпада за очување ресурса и поступак опоравка, квантификација угљеника у земљишту.

1.2.2 Дозиметрија

Дозиметрија се бави одређивањем дозе јонизирајућег зрачења, првенствено у људском ткиву, али и у разним другим материјалима. Неутронска дозиметрија је од велике важности за заштиту од зрачења у близини акцелератора честица и нуклеарних реактора, за квантификацију ефеката излагања космичком зрачењу, као и код терапије брзим и спорим неутронима.

Иако је равој нуклеарне и физике елементарних честица унео у употребу много врста детектора, они се заснивају на истом принципу: предаја дела или целокупне енергије зрачења детекторској маси, где се она преводи у неку другу форму енергије која је погодна за људско опажање.

За детекцију, неутралне честице најпре морају проћи кроз неку врсту реакције у детектору чији ће производ бити наелектрисане честице које јонизују или ексцитирају атоме материјала детектора. Облик у коме се појављује енергија зависи од детектора и његове конструкције. На пример, гасни детектори директно прикупљају јонизационе електроне за формирање струјног сигнала, док у сцинтилаторима ексцитација и јонизација доприносе појави молекуларних прелаза чији је крајњи резултат емисија светлости. У фотографским емулзијама јонизација изазива хемијске реакције које омогућују формирање латентне слике трага.

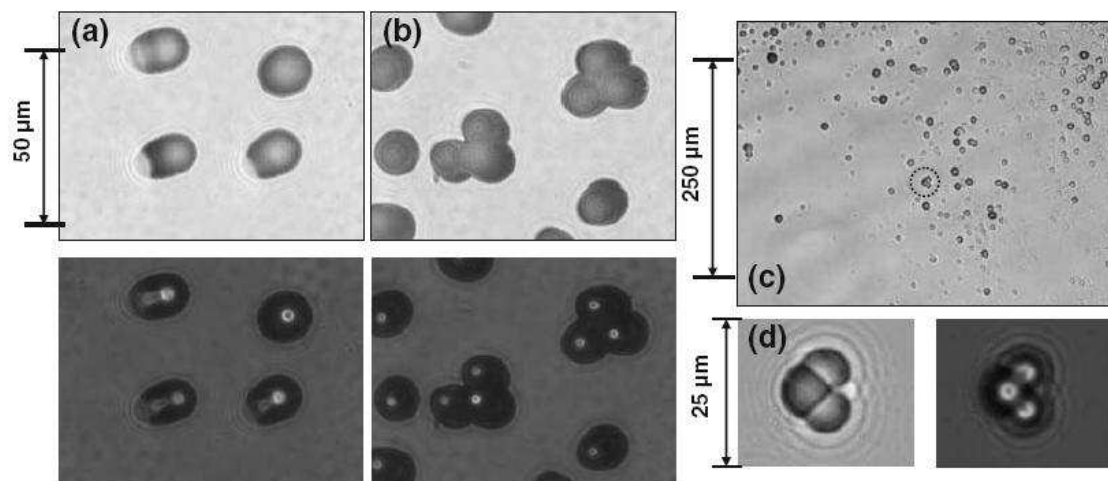
Савремени детектори су претежно електричне природе. У неком тренутку информација из детектора се претвара у електрични сигнал који је погодан за електронску обраду. Такви детектори користе активне методе за детекцију и дозиметрију неутрона.

Поред активних, користе се и пасивне методе, односно методе које не захтевају директно напајање енергијом у току излагања, Међу данас коришћеним пасивним личним неутронским дозиметрима, траг детектори се сматрају једном од најперспективнијих техника. Посебно место у свету траг детектора заузима *CR-39* детектор.

***CR-39* детектор**

Међу бројним предностима *CR-39* детектора у односу на друге мерне технике су велика осетљивост на протоне, нискоенергетски праг за неутроне, као и неосетљивост на гама зрачење. Због тога је *CR-39* најпогоднији кандидат за примену у личној неутронској дозиметрији.

При расејању неутрона на атомима *CR-39* детектора ($C_{12}H_{18}O_7$) производе се узмакнута језгра Н, С и О, док се у нуклеарним реакцијама стварају секундарне наелектрисане честице (α честице, протони, деутерони...). Секундарне наелектрисане честице и узмакнута језгра остављају оштећења при проласку кроз детекторски материјал, која се називају “латентни трагови”. Попречне димензије латентних трагова су до 10 nm, тако да се могу видети само под електронским микроскопом. Траг се може визуелизирати (учинити видљивим) под оптичким микроскопом, ако се делује агресивним хемијским агенсом, као што је на пример, водени раствор *NaOH* или *KOH*. Услед веће хемијске активности, раствор нагриза оштећени део више него неоштећени, тако да се латентни траг знатно увећава и може се посматрати обичним оптичким микроскопом. Примери нагрижених трагова су дати на слици.



Слика 1.3: Трагови настали озрачивањем *CR-39* детектора

1.3 Дефиниција проблема

Мерење густине трагова насталих траг детектором *CR-39* визуелном проценом броја трагова по јединици површине помоћу микроскопа траје пуно времена. Још један од изазова је и опис развоја трага, односно раст трагова. Ови процеси могу да трају и по више сати.

Основна хипотеза Мастер рада је да се овај посао може аутоматизовати и значајно убрзати применом алгоритама за обраду слика.

Као основни програмерски алати користе се решења отвореног кода: језик *C++* спрегнут са библиотеком *OpenCV* (*Open Source Computer Vision Library*), као и *Qt* фрејмворк за кориснички интерфејс.

Глава 2

Обрада слика употребом *OpenCV* библиотеке

OpenCV (*Open Source Computer Vision Library*) је библиотека отвореног кода која садржи имплементације више стотина алгоритама рачунарског вида (*computer vision*). Библиотека је написана у *C++* програмском језику, али подржава и итерфејсе ка *C*, *Python* и *MATLAB* програмским језицима, а у разоју су и интерфејси за *CUDA* и *OpenCL* језике. Подржана је на свим видећим оперативним системима: *Windows*, *Linux*, *Android* и *Mac OS*. Библиотека је објављена под *BSD* лиценцом, те је стога погодна и за академску и за комерцијалну употребу.

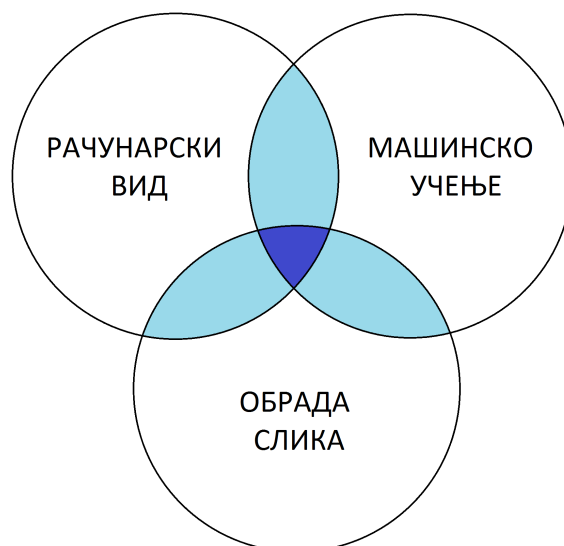
Имплементирани алгоритми се могу користити за препознавање облика, детекцију и препознавање лица, праћење покрета при видео снимку, спајање више слика у једну, препознавање маркера за проширену стварност и слично. Такви алгоритми су примењени у бројним програмима попут програма за видео надзор, навигацију и аутоматизацију рада робота, проверу производа у фабрикама, асистенцију при војњи аутомобила и тд.

Рачунарски вид помаже у прикупљању релевантних информација са слика и доношењу одлука базираним на тим подацима. Циљ рачунарског вида је да омогући да рачунар посматра ствари на исти начин као и људи. Основни кораци система базираног на рачунарском виду су:

- прикупљање слика,
- манипулација сликама,
- извлачење релевантних информација и
- доношење одлука.

Као што се види из наведеног, за један такав систем јакo су важни и алгоритми машинског учења (*machine learning*) као и алгоритми обраде слика (*image processing*). *OpenCV* библиотека садржи такве алгоритме. Компонента која нас занима је управо обрада слика.

Обрада слика је процес манипулације подацима слике у сврху прикупљања релевантних информација.



Слика 2.1: Однос рачунарског вида са машинским учењем и обрадом слика

2.1 Сиво скалирана слика

Први корак за већину алгоритама обрада слика је рачунање сиво скалиране слике на основу оригиналне слике, слике која садржи црвену, зелену и плаву компоненту. У даљем процесу користи се само сиво скалирана слика. На тај начин постиже се значајна уштеда у меморији јер слика која се обрађује може бити копирана више пута, самим тим постижу се боље перформансе, а добија се и на једноставности алгоритама.

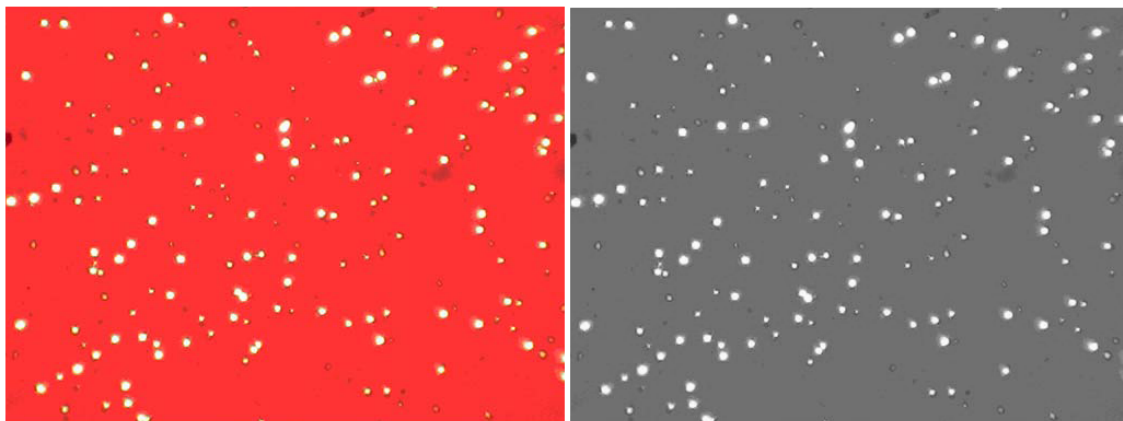
Изворни код 2.1: Рачунање сиво скалиране слике

```
1 // Load BGR image.
2 Mat bgr = imread(path, CV_LOAD_IMAGE_COLOR);
3
4 // Convert image to grayscale.
5 Mat grayscale;
6 cvtColor(bgr, grayscale, CV_BGR2GRAY);
```

Сиво скалирана слика рачуна се као:

$$Y \leftarrow 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \quad (2.1)$$

где су Y - сиво скалирана слика, R - црвена компонента оригиналне слике, G - зелена компонента оригиналне слике и B - плава компонента оригиналне слике. Коефицијенти представљају измерену перцепцију интензитета код трохроматских људи. Конкретно, људски вид је најосетљивији на зелену, а најмање на плаву боју.



Слика 2.2: Оригинална (лево) и сиво скалирана (десно) слика

2.2 Сегментација слике

Један од основних циљева обраде слика је идентификација објеката који се појављују на дигиталној слици. Задатак сегментације слике је да произведе поједностављене информације за анализу. Сегментација пружа информације о позицији и изгледу објекта.

Постоје различите методе сегментације, а избор конкретне методе се врши на основу примене. Неке од најчешћих метода сегментације су:

- Сегментација прагом (*Thresholding*) подразумева раздвајање објеката на основу боје пиксела. У зависности од боје, пиксел припада одређеној групи објеката. Уколико се врши сегментација са једним прагом, метода се назива бинаризација. Бинаризација се најчешће користи за раздвајање објеката од позадине. Ово је најједноставнији и најбржи метод сегментације. Сегментација прагом се примењује у алгоритмима препознавања текста за раздвајање текста од позадине.
- Сегментација кластеризацијом (*Clustering*) подразумева груписање пиксела сличних карактеристика у групе/објекте. Поступак сегментације се састоји од рачунања вектора одлика за сваки пиксел, проналажења карактеристичних вектора пиксела - центара кластера који свакако припадају различитим објектима и придруживање сваког пиксела једном од кластера. Ова метода се примењује са великим успехом у обради сателитских и авионских снимака, а највећа мана је велика рачунска сложеност.
- Детекција ивица (*Edge detection*) - подразумева проналажење тачака слике око којих се значајно мења интензитет боје. На овај начин добијају се линије које представљају ивице објеката. Пошто добијене линије нису увек спојене додатно се користе алгоритми за реконструкцију недостајућих сегмената. Детекција ивица је моћнији, али и спорији метод сегментације.
- Слив (*Watershed*). Свака слика се може посматрати као топографска површина где низак интензитет боје означава брда и врхове, а висок интензитет означава долине. Идеја алгоритма је да се свака изолована долина (локални минимум интензитета боје) пуни различито обојеном водом. Са порастом новог, вода из различитих долина почиње да се спаја. Да би се избегло спајање, постављају

се баријере на месту спајања. Вода наставља да расте све док сви врхови не буду под водом. Постављене баријере представљају резултат сегментације.

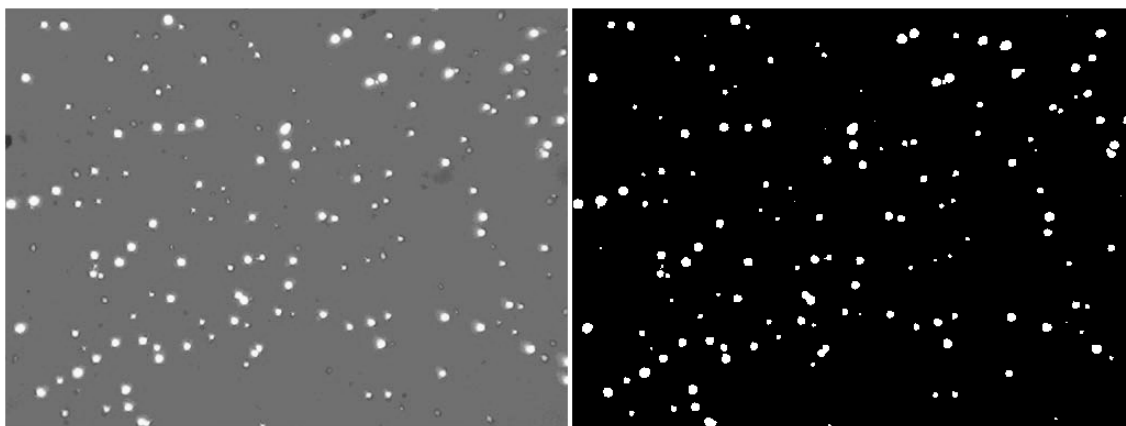
У сличају детекције трагова неутронске дозиметрије трагови су јасно представљени светлом бојом на тамној позадини, па је издвајање позадине једноставан задатак. Компликованији случај је раздвајање трагова који се преклапају без јасне границе између преклопљених трагова. За решавање овог проблема користи се модификована верзија *watershed* алгоритма. Основни кораци детекције трагова су:

- раздвајање трагова од позадине,
- проналажење трагова и
- сегментација трагова.

У случају да су трагови превише преклопљени, сегментација трагова не може да их детектује као засебне трагове. Из тог разлога оставља се могућност корекције трагова у каснијем раду.

2.2.1 Раздвајање трагова од позадине

За раздвајање трагова од позадине користи се сегментација са једним прагом, односно бинаризација. Бинаризација подразумева раздвајање светлих и тамних делова слике, односно у овом случају идентификовање светлих објеката на тамној позадини.

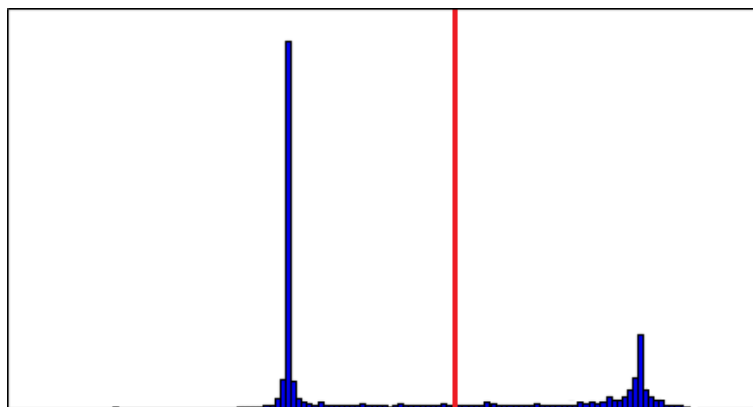


Слика 2.3: Сиво скалирана (лево) и бинарна/црно-бела (десно) слика

Улазни параметар за бинаризацију је сиво скалирана слика. Резултат бинаризације је црно-бела слика истих димензија као и сиво скалирана слика. Црна боја представља позадину, а белом бојом се означавају пиксели који припадају објектима. Вредност сваког пиксела резултујуће слике рачуна се као:

$$binary(x, y) = \begin{cases} white, & grayscale(x, y) > thresh, \\ black, & иначе. \end{cases} \quad (2.2)$$

где је *binary* - бинарна слика, *grayscale* - сиво скалирана слика, *thresh* - праг, *white* - вредност белог пиксела, *black* - вредност црног пиксела, а *x* и *y* одговарајући индекси пиксела слика.

Слика 2.4: Хистограм интензитета боје на сиво скалираној слици и *Otsu* праг

Изворни код 2.2: Рачунање бинарне слике са унапред задатим прагом

```

1 // Calculate binary image.
2 threshold(
3     grayscale, // Grayscale image.
4     binary,    // Binary image.
5     thresh,    // Threshold value.
6     255,       // Max pixel value, i.e. white pixel value.
7     CV_THRESH_BINARY);

```

Праг према коме се врши класификација може бити и аутоматски одређена вредност помоћу *Otsu* метода. Употребом овог метода креира се хистограм вредности пиксела. Према очекивању постоје две класе вредности хистограма (једна за тамне пикселе и једна за светле пикселе), а као оптимална граница узима се средња вредност између те две класе.

Изворни код 2.3: Рачунање бинарне слике са *Otsu* прагом

```

1 // Calculate binary image.
2 threshold(
3     grayscale, // Grayscale image.
4     binary,    // Binary image.
5     0,         // Threshold, ignored value.
6     255,       // Max pixel value, i.e. white pixel value.
7     CV_THRESH_BINARY | CV_THRESH_OTSU);

```

2.2.2 Проналажење трагова

У овом тренутку имамо раздвојене трагове од позадине, али трагови могу бити међусобно повезани. Да бисмо раздвојили и спојене трагове рачунамо матрицу удаљености од позадине (*distance transform*).

Матрица удаљености од позадине се обично рачуна само за бинарне слике. Резултат је сиво скалирана слика која личи на улазну слику, стим да интензитети боја тачака унутар објеката показују удаљеност до најближе границе са позадином.

Изворни код 2.4: Проналажење централних региона трагова

0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0

→

0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	0
0	1	2	2	2	2	2	1	0
0	1	2	3	3	3	2	1	0
0	1	2	3	3	3	2	1	0
0	1	2	2	2	2	2	1	0
0	1	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0

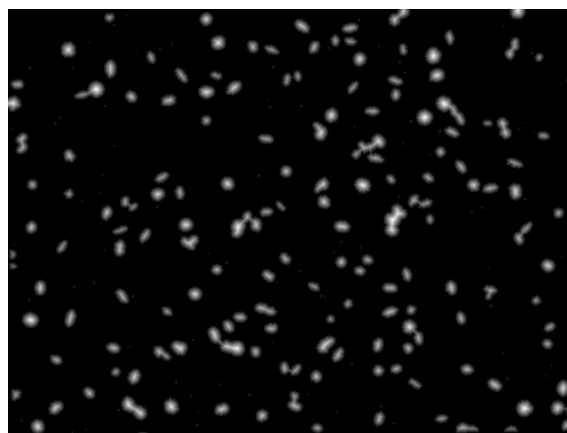
Слика 2.5: Матрица удаљености од позадине

```

1 // Calculate distance matrix.
2 cv::Mat dist;
3 cv::distanceTransform(binary, dist, CV_DIST_L2, 3);
4 cv::normalize(dist, dist, 0, 1., cv::NORM_MINMAX);
5
6 // Binarize distance matrix.
7 cv::threshold(dist, dist, .5, 1., CV_THRESH_BINARY);
8
9
10 // Create the CV_8U version of the distance image
11 // It is needed for cv::findContours().
12 cv::Mat dist_8u;
13 dist.convertTo(dist_8u, CV_8U);
14
15 // Find total markers
16 std::vector<std::vector<cv::Point> > contours;
17 cv::findContours(dist_8u, contours, CV_RETR_EXTERNAL,
18                 CV_CHAIN_APPROX_SIMPLE);
19
20 // Total objects.
21 int ncomp = contours.size();

```

На овај начин највеће вредности ће остати у централним деловима трагова. Применом бинаризације на новонасталу матрицу омогућује се раздавање трагова који се додирују.



Слика 2.6: Матрица удаљености

Означене делове нове бинарне слике третирамо као централне делове трагова.

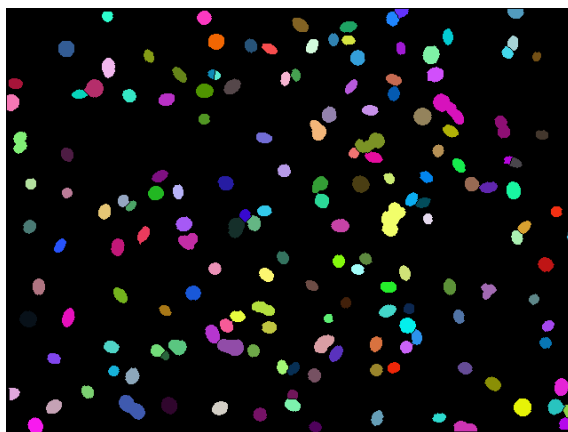
Користимо их као полазне регионе за *watershed* алгоритам.

2.2.3 Сегментација трагова

OpenCV садржи модификовану верзију *watershed* алгоритма. Имплементација не проналази аутоматски регионе из којих креће да означава сличну околину, већ те регионе тражи као улазни параметар.

Изворни код 2.5: Сегментација трагова

```
1 cv::Mat markers = cv::Mat::zeros(dist.size(), CV_32SC1);
2 for (int i = 0; i < ncomp; i++)
3 {
4     cv::drawContours(markers, contours, i, cv::Scalar::all(i+1), -1);
5 }
6
7 cv::watershed(binary, markers);
```

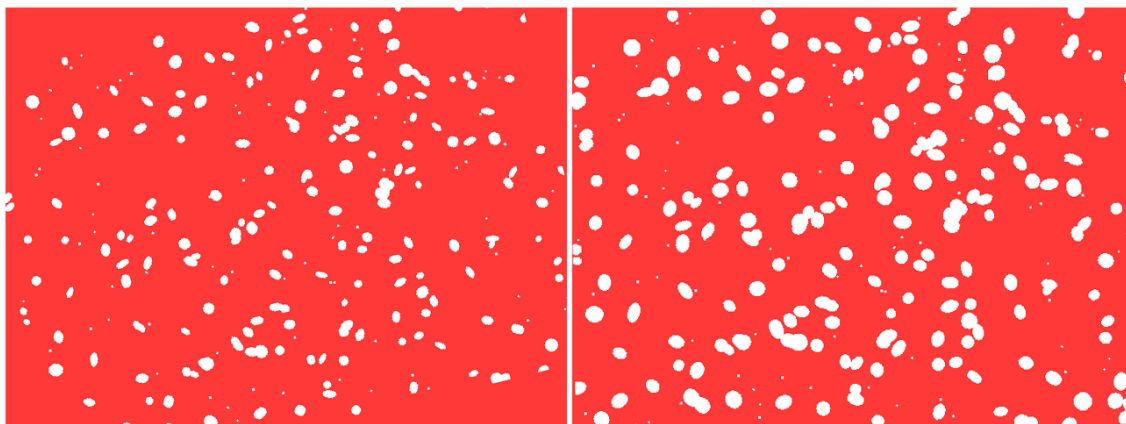


Слика 2.7: Детектовани трагови

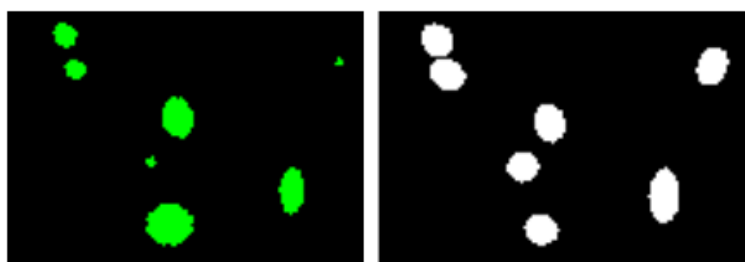
2.3 Исправљање ротације слике у односу на претходну слику из секвенце

При прикупљању секвенце слика у различитим временским тренуцима дешава се да материјал детектора није усликан под истим углом. У том случају позиција истих трагова на различитим сликама није иста и није могуће пратити раст трагова. Зато приступамо отклањању ротације (*deskew*) слике у односу на предходну слику из секвенце.

Пошто је могуће да се трагови не шире у истом правцу и на исти начин, може да се деси и да се контуре тргова разликују значајно. Из тог разлога за детекцију угла између слика не користе се оригиналне слике, већ потпис слика (*fingerprint*). Потпис слике са траговима неутронске дозиметрије се генерише након детекције трагова тако што се за сваки детектовани траг испртава нормализовани траг, односно елипса чији



Слика 2.8: Претходна слика из секвенце (лево) и нова слика (десно)



Слика 2.9: Препознати трагови (лево) и потпис (десно)

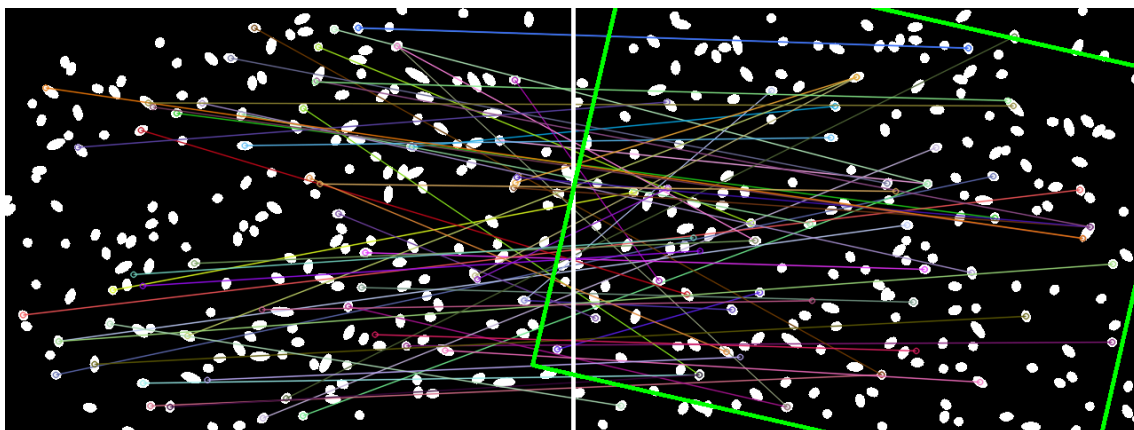
краћи пречник има предефинисану дужину, а дужи пропорционално већи пречник у односу на оригинално детектовани траг.

Проналажење сличности између две исте слике је део многих апликација рачунарског вида: калибрација камере, *3D* реконструкција, препознавање објеката и тако даље. Један од алгоритама за детекцију сличности између слика је *SURF* (*Speeded Up Robust Features*).

SURF алгоритам проналази интересне тачке (*key points*) на сликама и описује их описним вектором (*feature vector*). Интересне тачке су препознатљиве тачке објекта које се могу пронаћи при различитим погледима на објекат. Описивањем ових тачака на такав начин да су независне од величине слике, ротације и осветљења постиже се циљ уочавања сличности између различитих слика. Поређењем описних вектора може се донети закључак о трансформацији слике.

Алгоритам за утврђивање ротације слике се састоји из три главна корака:

- детекција интересних тачака,
- рачунање описног вектора у интересним тачкама и
- поређење описних вектора.



Слика 2.10: Потписи претходне (лево) и нове слике (десно) са одређеним преклапањем

2.3.1 Детекција интересних тачака

Детекција интересних тачака. Интересне тачке се бирају на карактеристичним местима, као што су оштре ивице, мрље, Т-спојеви.

Изворни код 2.6: Детекција интересних тачака

```
1 // Step 1: Detect the keypoints using SURF Detector.
2 int minHessian = 400;
3 SurfFeatureDetector detector(minHessian);
4
5 std::vector<KeyPoint> keypointsNew, keypointsPrev;
6 detector.detect(imgNew, keypointsNew);
7 detector.detect(imgPrev, keypointsPrev);
```

2.3.2 Рачунање описног вектора

Рачунање описног вектора у интересним тачкама. Описни вектор представља вектор који описује околину интересне тачке. Овај вектор мора бити карактеристичан, отпоран на шум и дефекте који настају при сликању.

Изворни код 2.7: Рачунање описних вектора

```
1 // Step 2: Calculate descriptors (feature vectors).
2 SurfDescriptorExtractor extractor;
3
4 Mat descriptorsNew, descriptorsPrev;
5 extractor.compute(imgNew, keypointsNew, descriptorsNew);
6 extractor.compute(imgPrev, keypointsPrev, descriptorsPrev);
```

2.3.3 Поређење описних вектора и одређивање ротације слике

На крају се врши упоређивање описних вектора и на основу њихове сличности закључује се о сличности међу сликама. Крајњи резултат је матрица хомографије из које се мог извући информације о трансформацији слике.

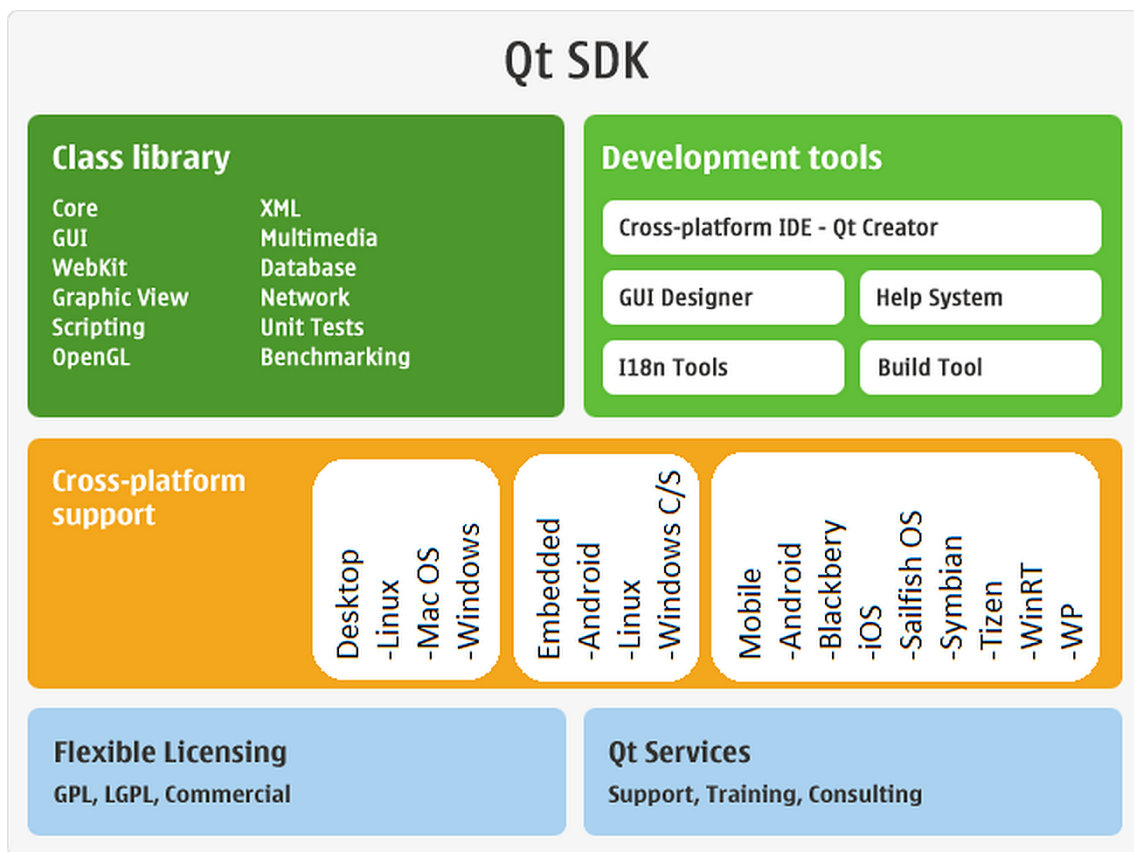
Изворни код 2.8: Поређење описних вектора и одређивање ротације нове слике

```
1 // Step 3: Matching descriptor vectors using FLANN matcher.
2 FlannBasedMatcher matcher;
3
4 std::vector< DMatch > matches;
5 matcher.match( descriptors_object, descriptors_scene, matches );
6
7 double maxDist = 0;
8 double minDist = 100;
9
10 //-- Quick calculation of max and min distances between keypoints.
11 for( int i = 0; i < descriptors_object.rows; i++ )
12 {
13     double dist = matches[i].distance;
14     if (dist < minDist) minDist = dist;
15     if (dist > maxDist) maxDist = dist;
16 }
17
18 // Get only "good" matches.
19 double goodMatchThreshold = min_dist + (max_dist - min_dist) / 2;
20 std::vector< DMatch > goodMatches;
21 for( int i = 0; i < descriptors_object.rows; i++ )
22 {
23     if(matches[i].distance < goodMatchThreshold)
24     {
25         goodMatches.push_back( matches[i] );
26     }
27 }
28
29 // Calculate homography.
30 std::vector<Point2f> goodNew;
31 std::vector<Point2f> goodPrev;
32
33 for( size_t i = 0; i < goodMatches.size(); i++ )
34 {
35     // Get the keypoints from the good matches.
36     goodNew.push_back(keypointsNew[goodMatches[i].queryIdx].pt);
37     goodPrev.push_back(keypointsPrev[goodMatches[i].trainIdx].pt);
38 }
39
40 Mat homography = findHomography(goodNew, goodPrev, CV_RANSAC);
```

Глава 3

Развој корисничког интерфејса употребом *Qt* фрејмворка

Qt је фрејмворк за развој корисничког интерфејса. Омогућује развој апликација које се могу употребити и за десктоп рачунаре и за мобилне телефоне без потребе да се поново пише изворни код. *Qt* је надскуп стандардног *C++* програмског језика тако да програмери могу да користе или *Qt* или стандардне *C++* библиотеке, а могу да се користе и у комбинацији.



Слика 3.1: *Qt SDK (Software Development Kit)*

Qt већ користе стотине хиљада програмера. Значајани делови *Linux KDE* дистрибуције, *Google Earth* и *Skype* програма су написани у овом фрејмворку. Обухвата богат

скуп графичких контрола *Widgets* које пружају стандардне *GUI* функционалности као што су менији, контекстни мени, *drag and drop*, и слично. Уводи иновативну и безбеднију алтернативу за комуникацију између објеката названу *Signals and Slots*, али подржава и конвенционални модел за реаговање на клик миша, притисак дугмета, и друге улазне догађаје.

Од самог почетка *Qt* је био доступан за слободну комерцијалну употребу. То је омогућило развој апликација без икаквих ограничења лиценцирања. Постојало је више различитих лиценци. У почетку изворни код је био под лиценцом *FreeQT* која је дозвољавала преузимање и употребу кода, али је објављивање модификација било забрањено. Тренутно, код је доступан под *GNU Lesser General Public 2.1 (LGPL)*, која га чини доступним за употребу у пројектима и отвореног и затвореног кода.

3.1 Развојни алати

Qt нуди широк скуп алата за развој софтвера.

- ***Qt Assistant*** је алат који графички представља *HTML* документацију пројекта. Садржи примере кода и пуно корисних информација о томе како се користите класе фрејмворка. Такође, пружа могућност једноставне претраге документације. Овај алат може да се прилагоди и да се објављује заједно са производом као додатак за графички приказ документације конкретног производа.
- ***Qt Designer*** је алат за пројектовање и изградњу корисничког интерфејса. Кориснички интерфејс се развија на *what you see what you get (WYSWYG)* начин. Могуће је тестирање понашања интересја при различитим резолуцијама, што олакшава развој корисничког интерфејса на различитим платформама. Визуелне контроле креиране са овим дизајнером интегришу се са програмским кодом употребом модела сигнала и слотова што олакшава раздвајање функционалности од корисничког интерфејса. Све карактеристике визуелних контроле се могу мењати и у коду динамички. Информације о корисничком интерфејсу се чувају у фајловима са екстензијом *.ui*. То су *XML* фајлови на основу којих се аутоматски генерише *C++* код који се даље компајлира.
- ***Qt Linguist*** је алат који омућава локализацију апликације.
- ***Qt Creator*** је развојно окружење (*IDE - Integrated Development Environment*) које интегрише претходно наведене алате. Такође, пружа алате за извршавање задатака током целог циклуса развоја апликације. Аутоматизује и убрзава задатке као што су креирање пројеката, додавање нових фајлова, нуди семантичко означавање, проверу синтаксе кода, комплетирање наредбе, рефакторисање, као и многе друге корисне функције.

3.2 Модули

Qt фрејмворк је подељен у неколико засебних модула. Да би се користиле функционалности одређеног модула потребно је додати одговарајућу променљиву у пројекти фајл. На пример, за употребу класа за рад са *XML* фајловима потребно је додати:

Изворни код 3.1: Укључивање модула за рад са XML фајловима

```
1 QT += xml;
```

Коришћени модули за приказ корисничког интерфејса у софтверу за детекцију трагова неутронске дозиметрије обухватају:

- **QtCore** садржи основне класе које користи свака Qt апликација, као и други модули. Овај модул омогућава рад са објектима (*QObject*), стринговима (*QString*), фајловима (*QFile*), фолдерима (*QDir*), локализацијом (*QLocale*) и тако даље. Овај модул је подразумеван, тако да га није потребно експлицитно укључивати.
- **QtGui** модул садржи компоненте графичког корисничког интерфејса (*GUI*). Садржи специјализоване контроле *Widgets* (*QCalendarWidget*, *QDockWidget*), контејнере (*QGroupBox*, *QStackedWidget*) и дијалоге (*QFileDialog*, *QPrintDialog*).
- **QtXml** садржи класе за рад са *Xml* фајловима. Омогућава читање, писање и форматирање *Xml* структура.

3.3 Сигнали и слотови

Сигнали и слотови (*Signals and Slots*) су основни механизам свих Qt програма. Омогућавају комуникацију између објеката, који не знају ништа једни о другима. Типичан пример је када желимо да промена у једној од контрола утиче на промене у другој контроли. Само наслеђене класе из *QObject* могу применити механизам сигнала и слотова. *Widget* контроле имају пуно предефинисаних сигнала и слотова, али је могуће додефинисати нове по потреби.

Идеја је да један објекат може да пошаље сигнал о са информацијама насталој промени без знања да ли ће ту информацију неко прихватити. Други објекти се могу повезати на ту врсту сигнала, и при емитовању сигнала од стране првог објекта могу извести одређену акцију. Могуће је повезати само сигнале и слотове са истим параметрима. Слотови се коористе за пријем сигнала, али у суштини то су обичне C++ функције тако да се могу извршити у било које време.

Изворни код 3.2: Дефиниција и емитовање сигнала

```
1 class Sender : public QObject
2 {
3     Q_OBJECT
4
5 public:
6     void sendSignal()
7     {
8         // Emit signal.
9         emit newMessage("Message text!");
10    }
11
12 signals:
13     void newMessage(QString message);
14 };
```

Изворни код 3.3: Дефиниција слота

```
1 class Receiver : public QObject
2 {
3     Q_OBJECT
4
5     public slots:
6         receiveMessage(QString message)
7         {
8             // Print received message to debug console.
9             qDebug() << message;
10        }
11 };
```

Сигнали и слотови се могу повезати на следећи начин:

Изворни код 3.4: Повезивање сигнала и слотова

```
1 QObject::connect(
2     sender,
3     SIGNAL(newMessage),
4     receiver,
5     SLOT(receiveMessage));
```

На један сигнал могуће је прикачити више слотова. Чекање на сигнал се може и укинути:

Изворни код 3.5: Укидање везе сигнала и слота

```
1 QObject::disconnect(
2     sender,
3     SIGNAL(newMessage),
4     receiver,
5     SLOT(receiveMessage));
```

Глава 4

Софтвер за детекцију трагова неутронске дозиметрије

Чему слижи софтвер.
Коме је намењен.
Портабиалан.
Лак за коришћење.

4.1 Структура пројекта

Репозиторијум.
Организација пројекта.
Дијаграм класа.
Опис задужења сваке класе.

4.2 Начин употребе

Класично упутство за употребу. Слика алата са означеним основним деловима корисничког интерфејса. По реченица о сваком делу, касније опширније.

4.2.1 Управљање пројектима

(у левом делу корисничког итерфејса)
Креирање пројекта.
Опције пројекта.
Додавање слике.
Опције слике.

4.2.2 Управљање прегледом слике

(Горњи, доњи и централни део корисничког итерфејса)
Одабир прегледа слике (оригинална, сиво скалирана, бинарна и тд.)
Манипулације сликом (зумирање).
Промена боја елемената на проказаној слици.

4.2.3 Приказ и извоз резултата обраде

(десни део корисничког итерфејса + мени)
Приказ и објашњење контроле са резултатима.
Пример ЦСВ фајла који се експортује.
Извоз тренутног приказа.

4.2.4 Корекција обраде

Аутоматски резултати нису савршени, потребне су интервенције.

Подешавање параметара обраде слике

Подешавање прага бинаризације.
Примена филтера: блур и тд.

Корекција детектоване ротације слике

Како се врши корекција угла.

Корекција детектованих трагова

Корекција спојених трагова који нису раздвојени.

Глава 5

Закључак

Анализа.

Примена.

Шта је добро.

Шта може боље.

Како се у будућности оже унапредити софтвер.

Библиографија

- [1] James Turner, *Atoms, Radiation, and Radiation protection*. Wiley-VCH, New York 2007.
- [2] <http://opencv.org>, *OpenCV* званична веб страна
- [3] <http://qt-project.org/>, *Qt* званична веб страна