



UNIVERSITÀ POLITECNICA DELLE MARCHE
FACOLTÀ DI INGEGNERIA

Corso di Laurea in Ingegneria Biomedica

**Algoritmo per il tracciamento di cellule
tumorali in vitro da immagini di microscopia
ottica**

Algorithm for tracking of cancer cells in vitro from optical microscope images

Relatore: Chiar.mo
Prof. Francesco Piva

Tesi di Laurea di:
Luca Lorenzetti

A.A. 2019/2020

Indice

INTRODUZIONE	p.3
CAPITOLO 1 – LOCOMOZIONE CELLULARE	p.3
<i>1.1 Chemiotassi</i>	p.4
CAPITOLO 2 - TUMORI	p.5
<i>2.1 Generalità sui tumori</i>	p.5
<i>2.2 Principi di proliferazione cellulare</i>	p.6
<i>2.3 Metastatizzazione</i>	p.7
<i>2.4 Transizione epitelio-mesenchimale</i>	p.9
<i>2.5 Modificazioni epigenetiche</i>	p.11
CAPITOLO 3 – MICROSCOPIO OTTICO	p.13
<i>3.1 Microscopio a contrasto di fase</i>	p.14
<i>3.2 Microscopia a fluorescenza</i>	p.15
<i>3.3 Microscopio a scansione confocale</i>	p.15
CAPITOLO 4 – PROCEDURA ALGORITMO	p.17
<i>4.1 Descrizione delle immagini ottenute al microscopio</i>	p.17
<i>4.2 Elaborazione delle cellule</i>	p.18
<i>4.3 Centroidi</i>	p.19
<i>4.4 Rilevazione dello spostamento cellulare</i>	p.20
<i>4.5 Elaborazione delle cellule in fase mitotica</i>	p.21
<i>4.6 Velocità e direzione delle cellule</i>	p.23

<i>4.7 Interfaccia grafica (GUI)</i>	p.24
<i>4.8 Codice algoritmo</i>	p.26
CAPITOLO 5 - CONCLUSIONE	p.77
BIBLIOGRAFIA	p.78

INTRODUZIONE

CAPITOLO 1 - LOCOMOZIONE CELLULARE

La locomozione cellulare, quando avviene all'interno di un organismo pluricellulare, è difficile da osservare perché avviene nell'ambito di tessuti opachi da parte di cellule indistinguibili dalle cellule circostanti. Di conseguenza, la maggior parte dei ricercatori che studiano la locomozione cellulare si è indirizzata verso sistemi in cui si possono seguire singole cellule che si spostano sul fondo di una capsula di coltura. La locomozione di singole cellule di solito non è accompagnata da un evidente flusso citoplasmatico, ma è legata a vari tipi di espansioni della superficie cellulare al margine guida della cellula. Una rapida occhiata ad un fibroblasto di mammifero che si muove sulla superficie di una capsula di coltura serve come esempio di un tipo di locomozione molto diverso da quello di un'ameba. Quando si muove, un fibroblasto si appiattisce in modo caratteristico aderendo al substrato e assumendo una forma a ventaglio, con una estremità anteriore espansa e una "coda" stretta. Il suo movimento è irregolare, a volte in avanzamento e a volte in arretramento. Come faccia il fibroblasto a muoversi, si può capire esaminando il suo margine guida, che si estende all'esterno della cellula come una espansione ampia, appiattita detta lamellipodio, che scivola in avanti sopra il substrato. I lamellipodi sono di solito privi di strutture solide e il margine esterno presenta spesso un movimento ondulatorio che gli fa assumere un aspetto increspato. Quando un lamellipodio si espande dalla cellula, aderisce al substrato sottostante in punti specifici, fornendo dei siti temporanei di ancoraggio perché la cellula proceda oltre. L'espansione di un lamellipodio è associata all'assemblaggio di monomeri di actina in filamenti e alla organizzazione dei filamenti in sistemi organizzati mediante legami con proteine che legano actina. L'organizzazione tridimensionale dei microfilamenti del margine guida è dimostrata in modo ottimale dopo estrazione delle cellule con detergenti non ionici. I microfilamenti del lamellipodio forniscono una parte essenziale del meccanismo richiesto per la locomozione¹.

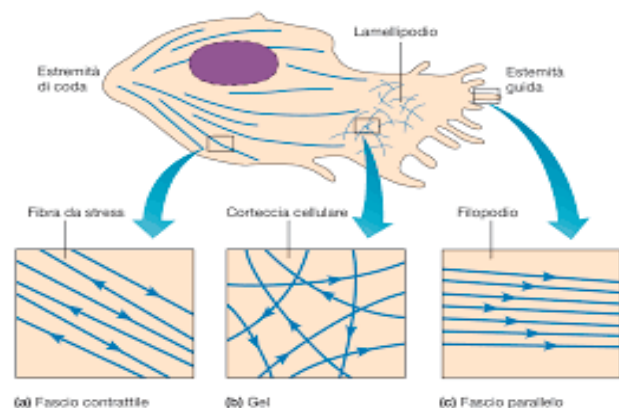


Figura 1 Locomozione cellulare.

1.1 Chemiotassi

Lo spostamento dipende da nutrienti e sostanze chimiche che lo influenzano. Per chemiotassi s'intende la locomozione delle cellule in risposta a segnali di natura chimica, o fattori chemiotattici, secreti da altre cellule nell'ambiente extracellulare. La chemiotassi gioca un ruolo essenziale in una grande varietà di cellule, dai procarioti ai vertebrati. A livello eucariotico, gli esempi meglio noti di chemiotassi sono i neutrofili e le cellule di *Dictyostelium*. Entrambi questi tipi di cellule, se esposti ad un gradiente chemiotattico, si orientano e si muovono rapidamente su un substrato. La risposta chemiotattica comporta la capacità della cellula di: (1) avvertire il gradiente chemiotattico, (2) polarizzarsi in direzione del gradiente e (3) coordinare i propri movimenti per migrare verso la fonte dello stimolo chemiotattico. A livello del citoscheletro ciò comporta l'esecuzione dei processi precedentemente descritti.

Lo spostamento cellulare e la chemiotassi sono alla base dell'invasività e migrazione di cellule tumorali metastatizzanti. Partiamo dalle fondamenta della malattia tumorale².

CAPITOLO 2 - TUMORI

2.1 Generalità sui tumori

Un **tumore** è una proliferazione monoclonale incontrollata (svincolata da qualsiasi meccanismo di controllo) di una popolazione cellulare, esprime gradi variabili di fedeltà rispetto al suo precursore. La teoria prevalente, formulata alla metà del secolo scorso, interpreta il cancro come un insieme di circa 200 malattie caratterizzate da un'abnorme crescita cellulare, svincolata dai normali meccanismi di controllo dell'organismo.

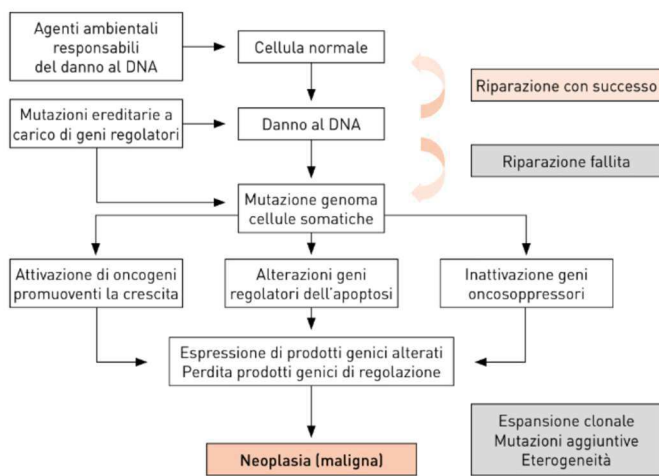


Figura 2 Cancerogenesi.

L'evento cancro richiede più di una mutazione a carico di diverse classi di geni. La perdita del controllo della proliferazione ha luogo solo in seguito a mutazioni nei geni che controllano la divisione cellulare, la morte cellulare e i processi di riparazione del DNA. Il nostro organismo è in grado, attraverso processi di

riparazione e attivazione del sistema immunitario, di contrastare i processi di trasformazione ma, quando questa capacità viene meno, la cellula si trasforma, attraverso varie tappe, in cellula tumorale. Sono necessarie quindi sia l'attivazione dei geni che promuovono la crescita (oncogeni) sia l'inattivazione dei geni che inibiscono la crescita (oncosoppressori). La cancerogenesi è un processo lungo e complesso: raramente una singola alterazione genetica è sufficiente per lo sviluppo del tumore. In genere un agente cancerogeno agisce sul DNA cellulare e provoca un processo di Iniziazione (rapido e irreversibile), seguito da una fase di Promozione della crescita neoplastica (lenta e irreversibile). Altri fattori devono intervenire per favorire la Progressione della malattia: nella maggior parte dei casi questi processi richiedono diversi anni. Altri meccanismi sono essenziali per la crescita tumorale, uno è rappresentato dal cosiddetto microambiente (cioè tutto ciò che ruota attorno al tumore): cellule favorenti, fattori di crescita, ma anche cellule che mangiano o uccidono la cellula tumorale. Un altro meccanismo è poi necessario alla progressione

della malattia: il tumore potrebbe infatti rimanere dormiente per molti anni finché non si accende una lampadina. È il cosiddetto switch angiogenico, cioè la capacità del tumore di costruire i propri vasi sanguigni tali da permettergli di crescere indisturbato³.

2.2 Principi di proliferazione cellulare

Il periodo di raddoppiamento è il tempo minimo nel quale una popolazione cellulare si moltiplica una volta (quello più rapido è quello della mucosa duodenale, di 1,6 giorni). Considerando che nei tumori le cellule proliferanti sono una parte ridotta (altre sono in G0, altre sono morte e altre sono differenziate e non si replicano), mediamente un tumore solido ha un tempo di replicazione di 100/120 gg, cioè occorrono 3/4 mesi affinché si abbia una replicazione cellulare.

La crescita di un tumore è:

- nel modello sperimentale di tipo esponenziale,
- nei tumori solidi umani (in vivo) di tipo gompertziana, si osserva cioè una proliferazione con decremento di crescita in funzione del tempo per fattori legati al tumore e all'uomo.

Le fasi di sviluppo tumorale sono:

- undetectable cancer: tumore non visibile;
- detectable cancer: il tumore arriva a dimensioni tali (in genere 1 cm) da renderlo riconoscibile dal punto di vista strumentale/clinico;
- host death: quando la malattia è conclamata abbiamo la possibilità di arrivare fino a 10^{12} cellule che corrispondono circa a 1 kg di tumore che è il massimo teoricamente compatibile con la vita.

Quindi, considerando che un nodulo di 0,5 cm (al di sotto del potere di risoluzione delle nostre metodiche) arriva a 1 cm nell'arco di 3/4 mesi,

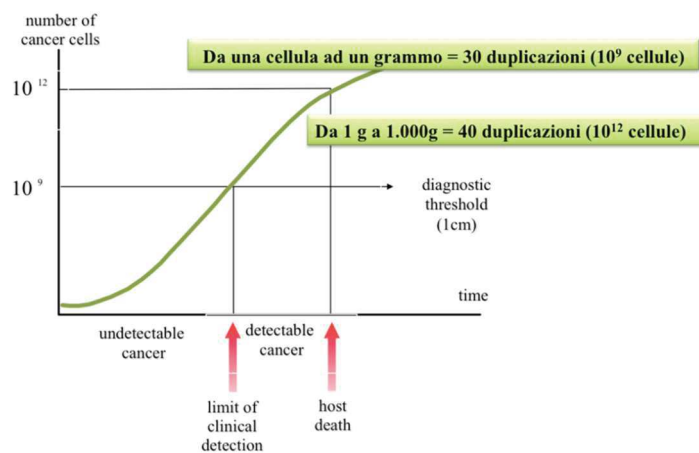


Figura 3 Proliferazione cellulare.

generalmente occorrono circa 5 anni da quando si forma la prima cellula alla formazione di un nodulo di 1 cm.

La proliferazione può inoltre essere:

- iperplasia: aumento del numero di cellule;
- ipertrofia: aumento del volume delle cellule;
- metaplasia: sostituzione di una cellula adulta con un'altra cellula adulta di altro tipo;
- displasia: alterazione nel volume, forma e organizzazione delle cellule.

Le prime tre sono reversibili, la displasia lo è più raramente e molto spesso è un precursore del cancro.

Nel passaggio displasia-cancro c'è una fase intermedia che è rappresentata dal tumore in situ: un tumore che non supera la membrana basale. Questo è importante perché al di sotto della membrana basale ci sono i vasi sanguigni e linfatici, di conseguenza solo un tumore che abbia superato questa membrana è in grado di dare metastasi linfonodali e a distanza⁴.

Il parametro che viene preso principalmente in considerazione per analizzare la proliferazione tumorale è rappresentato dal MIB-1/Ki-67. Il ki67 è una proteina nucleare non istonica espressa da un gene presente sul braccio lungo del cromosoma 10 di cellule in proliferazione, che in laboratorio viene analizzata mediante immunistochemica, utilizzando specifici anticorpi monoclonali come appunto il MIB-1. Essa rappresenta un indice di proliferazione della neoplasia in questione^{5,6}.

2.3 Metastatizzazione

La metastasi tumorale (o lesione secondaria o ripetitiva) è la riproduzione, nel medesimo organismo, del processo neoplastico, per distacco e migrazione a distanza di elementi cellulari e loro fissazione e moltiplicazione nella nuova sede d'impianto. I fattori che influenzano la cancerogenesi, e in particolare la metastatizzazione, soprattutto di pertinenza dell'ospite sono:

- fattori anatomici: la vascolarizzazione spiega la maggior frequenza di metastasi a livello epatico e polmonare,

- microambiente dell'organo: ci sono tanti studi che vanno a valutare le cellule tumorali circolanti. Facendo un prelievo di sangue io trovo cellule tumorali circolanti in qualsiasi paziente oncologico, sia in quello con malattia localizzata sia, a maggior ragione, in quello con tumore infiltrante. Però alcuni pazienti svilupperanno metastasi e altri no. Quindi ci sono dei fattori che determinano la metastatizzazione: enzimi proteolitici e altri fattori in grado di lisare la parete ed entrare nell'organo, microambiente dell'organo adatto. I pazienti (le malattie) che presentano questi fattori svilupperanno metastasi, gli altri no,

- fattori angiogenetici,
- risposta immunitaria dell'ospite: l'immunodepressione è un fattore di rischio per lo sviluppo dei tumori e il fatto che l'immunoterapia sia in espansione lo conferma.

Le vie di disseminazione di un tumore sono:

- per contiguità: è tipica delle patologie che crescono localmente e che recidivano localmente, uno di questi è il melanoma che spesso dà metastasi per contiguità o metastasi cutanee che si sviluppano per invasione dei tessuti vicini;
- per via ematica: è condizionata dalla vascolarizzazione dell'organo (per esempio la principale sede di metastasi di carcinoma del colon è il fegato),
- per via linfatica: ai linfonodi locoregionali (N) o a distanza (M),
- per via celomatica: nei tumori addominali si può avere la colonizzazione del peritoneo quindi la carcinosi peritoneale, difficilmente diagnosticabile anche alla TC (potremmo vedere: Ispessimento del foglietto del peritoneo, franco versamento ascitico, nodulazioni).

Recentemente un contributo alla conoscenza del processo di metastatizzazione è arrivato dal saggio wound-healing assay, un saggio di laboratorio utilizzato per studiare la migrazione cellulare e l'interazione cellula-cellula. Questo è anche chiamato "scrath assay" perché viene fatto "graffiando" un monostrato cellulare e catturando immagini a intervalli regolari con il microscopio time lapse.

Il saggio wound-healing assay è stato eseguito in piastre 12-well per colture cellulari. Dopo 24 ore dalle infezioni adenovirali le cellule sono state parzialmente rimosse eseguendo un taglio del monostrato con un puntale sterile per pipette (100µl). Successivamente le cellule sono state lavate con PBS e riposte in mezzo di coltura DMEM + FBS 2% in presenza o meno

dell'agonista β 2AR selettivo, fenoterolo (fenoterolo idrobromide, Sigma Aldrich, USA) $10\mu\text{M}$ e successivamente fissate in paraformaldeide 3,7%. In seguito alla fissazione le cellule sono state colorate con crystal violet 0,1%. ed osservate mediante l'utilizzo di un microscopio invertito Nikon TE2000-U connesso ad una fotocamera Nikon. Da ogni immagine acquisita è stata quantificata la migrazione cellulare misurando la distanza tra le estremità del taglio in 10 punti scelti casualmente utilizzando il software ImageJ. I dati ottenuti sono stati normalizzati rispetto all'ampiezza del taglio delle cellule di controllo fissate subito dopo la rimozione meccanica di cellule dal monostrato

Il saggio in questione è un ottimo strumento per studiare la migrazione cellulare poiché questo meccanismo è coinvolto in diversi aspetti fisiologici. La migrazione cellulare svolge un ruolo importante nella riepitelizzazione della pelle e quindi lo studio della migrazione cellulare può fornire indizi per la comprensione delle ferite non cicatrizzanti. La migrazione cellulare è anche fondamentale nei processi di sviluppo come la gastrulazione e l'organogenesi. La migrazione cellulare è anche coinvolta nelle risposte immunitarie e nelle metastasi del cancro. Con i recenti progressi tecnologici, questo test sta diventando molto utile soprattutto nel regno della biologia del cancro. È stato condotto uno studio per comprendere meglio il ruolo che claudin-7, una famiglia di proteine a giunzione stretta, svolge nella migrazione delle cellule tumorali umane del polmone. Il più lento tasso di migrazione delle cellule knockdown per claudin-7 supporta l'idea che questa proteina sia importante nella migrazione cellulare e quindi nella capacità di metastatizzare. In conclusione, le cellule effettuano una migrazione a causa di una moltitudine di segnali e meccanismi atti a chiudere una ferita; processo che si ritiene essere simile ai meccanismi coinvolti nella genesi delle metastasi⁴.

2.4 Transizione epitelio-mesenchimale

La caratteristica più importante che distingue un tumore benigno da uno maligno è la capacità di quest'ultimo di invadere la membrana basale, lo stroma e la parete dei vasi e di disseminarsi in sedi secondarie.

Il processo metastatico avviene mediante varie fasi successive che si possono riassumere in: perdita delle interazioni intercellulari e cellula-matrice, aumento della motilità e dell'invasività, ingresso e sopravvivenza nel torrente circolatorio e, infine, uscita dai vasi e "homing" e crescita nel nuovo tessuto. Perché queste fasi abbiano luogo è necessario che nella cellula tumorale, a causa delle mutazioni accumulate durante la progressione tumorale, vengano deregolati numerosi geni come detto precedentemente².

Le cellule tumorali, in generale, e quelle metastatiche in particolare, presentano una ridotta adesione intercellulare (perdita di inibizione da contatto). In numerosi casi, durante le varie fasi della progressione tumorale l'espressione della E-caderina va incontro a una progressiva riduzione. Sono stati documentati vari meccanismi responsabili della mancata espressione della caderina quali la presenza di mutazioni inattivanti, di silenziamento genico o di alterata degradazione. Per realizzare il programma invasivo, le cellule tumorali devono acquisire caratteristiche di motilità e vanno incontro ad un drastico cambiamento di morfologia e di profilo trascrizionale che rende le cellule epiteliali simili a quelle mesenchimali; questo processo è definito transizione epitelio-mesenchimale o EMT (Epithelial Mesenchymal Transition). La transizione EMT non è peculiare delle cellule tumorali, ma avviene normalmente in alcuni contesti fisiologici nell'organismo adulto (durante la riparazione delle ferite) o durante lo sviluppo embrionale (gastrulazione). Durante l'EMT, oltre alla perdita di espressione della E-caderina, si perde anche l'espressione di proteine dei filamenti intermedi (citocheratine) e la caratteristica polarità della cellula epiteliale. Le cellule acquisiscono una morfologia più simile

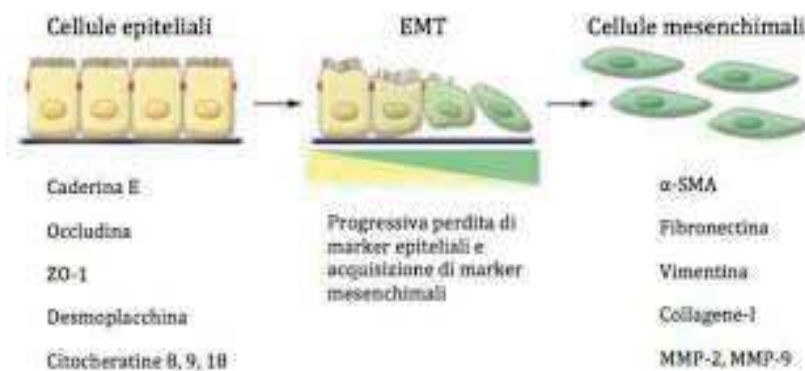


Figura 4 Transizione epitelio-mesenchimale.

a quella di un fibroblasto, con un aumento della motilità e dell'invasività. Aumenta anche l'espressione di diversi geni caratteristici delle cellule mesenchimali quali la vimentina (proteina dei

filamenti intermedi) e la N-caderina (tipica delle giunzioni aderenti delle cellule mesenchimali). In queste cellule, inoltre, aumenta l'espressione e secrezione di fibronectina e la secrezione di metallo proteasi, essenziali per permettere alla cellula di degradare la rete di proteine che costituiscono la membrana basale e la matrice extracellulare¹.

Ma quali fattori determinano l'attivazione della EMT in una cellula tumorale? Il meccanismo molecolare non è ancora completamente chiarito ma è sicuramente condizionato dal microambiente del tumore e, in particolare modo, dalla carenza di ossigeno. Infatti, l'ipossia è in grado di modulare l'espressione genica nelle cellule tumorali, stimolando la produzione di fattori trascrizionali. L'analisi mediante la tecnica dell'immunoistochimica rivela che l'espressione delle proteine coinvolte nella transizione EMT si può osservare lungo il margine invasivo del tumore che si sta infiltrando nel tessuto stromale. Le alterazioni nelle proprietà

adesive della cellula tumorale coinvolgono anche le integrine che connettono la cellula alla membrana basale/matrice extracellulare. Altre integrine sono coinvolte nelle fasi più tardive del processo di disseminazione e sono importanti nel regolare l'adesione delle cellule tumorali circolanti ai vasi. Motilità ed invasività rappresentano probabilmente le proprietà più importanti per una cellula tumorale che deve metastatizzare. Questo processo coinvolge necessariamente anche un dinamico riarrangiamento del citoscheletro di actina e miosina e le interazioni con la matrice extracellulare e, come già accennato, anche la degradazione della matrice mediante proteolisi localizzata.

Dopo l'invasione della matrice, le cellule tumorali devono disseminarsi attraverso la circolazione sanguigna e linfatica, uscire dal vaso e colonizzare un altro organo. Nel nuovo contesto stromale le cellule iniziano a crescere (micrometastasi) e attivano un processo detto di transizione mesenchimo-epiteliale (CMET - Mesenchimal-Epithelial Transition) che comporta la riacquisizione di caratteristiche proprie delle cellule epiteliali. In questo nuovo assetto la metastasi può accrescersi nel nuovo tessuto e colonizzarlo. Sembra che il passaggio da EMT a MET sia dovuto al fatto che le cellule ora si trovano in un contesto tissutale diverso da quello originale e quindi privo di quei fattori presenti nello stroma della sede primaria del tumore che avevano contribuito alla transizione EMT.

Molto rimane ancora da fare per comprendere la biologia della metastasi; in particolare rimane ancora oscuro il meccanismo che guida le cellule di un particolare tumore a colonizzare un organo anziché un altro. Ciò che chiaramente emerge, studiando il processo metastatico, è che per sopravvivere ed evolvere in un ambiente che gli è naturalmente ostile, il tumore cerchi di utilizzare strategie normalmente impiegate anche dalle cellule non tumorali. A questo proposito va nuovamente ricordato che molto di quello che si osserva e si studia nella cellula invasiva è straordinariamente simile a quello che accade durante le prime fasi dello sviluppo di un organismo (es. durante rimpianto nell'utero o nella gastrulazione); la differenza fondamentale tra i due processi sembra risiedere nel fatto che la cellula metastatica non è però più sensibile ad un efficace controllo da parte dell'ambiente⁴.

2.5 Modificazioni epigenetiche

L'inizio e la progressione del cancro sono controllati da eventi sia genetici che epigenetici. La complessità della carcinogenesi non può essere spiegata dalle sole alterazioni genetiche ma è dovuta anche a cambiamenti epigenetici. Quando si parla di epigenetica ci si riferisce allo studio dei meccanismi che alterano l'espressione genica senza alterare la sequenza del DNA primario.

I meccanismi epigenetici sono ereditabili e reversibili e includono cambiamenti nella metilazione del DNA, modificazioni dell'istone e piccoli microRNA non codificanti (miRNA). Tra questi ultimi è importante ricordare che la coda dell'istone costituisce quasi la metà della sequenza dell'istone, sporge dal nucleosoma e può essere metilata, acetilata, deacetilata e fosforilata secondo processi reversibili con funzione regolativa. Il processo di metilazione invece è responsabile della chiusura della cromatina rendendola meno accessibile, formando così eterocromatina; questo processo prevede l'avvicinamento di cariche elettriche opposte localizzate nelle code istoniche che attraendosi tendono a compattare la cromatina. L'interruzione dei processi epigenetici può portare ad un'alterata funzione genica e alla conseguente trasformazione cellulare maligna. Le alterazioni epigenetiche aberranti si verificano probabilmente in una fase molto precoce dello sviluppo neoplastico e sono ampiamente descritte come attori essenziali nella progressione del cancro².

Nei tumori umani, è noto che l'epigenomica aberrante contribuisce a varie fasi dello sviluppo neoplastico tra cui iniziazione, promozione, invasione, metastatizzazione e resistenza alla chemioterapia. Recentemente è stato proposto che oltre 300 geni e prodotti genetici siano epigeneticamente alterati in vari tumori umani.

Tuttavia, è evidente che alterazioni genetiche discrete nelle sole cellule neoplastiche non possano spiegare la carcinogenesi a più fasi in base alla quale le cellule tumorali sono in grado di esprimere diversi fenotipi durante le complesse fasi di sviluppo e progressione del tumore. In effetti, le cellule tumorali hanno un epigenoma alterato rispetto ai tessuti da cui derivano. Meccanismi epigenetici deregolati possono innescare instabilità genetica, con conseguente acquisizione di mutazioni genetiche nei geni soppressori del tumore e attivazione di mutazioni genetiche negli oncogeni³.

Curiosità: il ricordo è scritto nelle code degli istoni, di conseguenza possiamo mutarlo modificandole. Questi cambiamenti sono influenzati anche da fattori ambientali esterni che ne stimolano il rimodellamento aggiungendo o rimuovendo informazioni. Dati recenti dimostrano che i processi epigenetici svolgono un ruolo nel consolidamento della memoria ed aiutano anche a trasmettere le memorie acquisite attraverso le generazioni in maniera "lamarckiana". Sono stati trovati alcuni farmaci che hanno come target il macchinario epigenetico e riescono quindi a potenziare la funzione di memoria nei roditori e migliorare i fenotipi di malattia in modelli di patologie cerebrali come morbo di Alzheimer, corea di Huntington, depressione o schizofrenia.

CAPITOLO 3 - MICROSCOPIO OTTICO

Una sorgente di luce, che può essere esterna al microscopio o inserita all'interno della sua base, è necessaria per illuminare il campione. Il condensatore, posto sotto il tavolino portaoggetto, è necessario per concentrare i raggi diffusi dalla sorgente di luce e per illuminare il campione con un piccolo cono di luce sufficientemente intenso da permettere di vedere ogni piccola parte del campione, una volta ingrandita. Il modo in cui il campione è illuminato può avere un'importanza critica nel determinare la qualità dell'immagine ottenuta. I

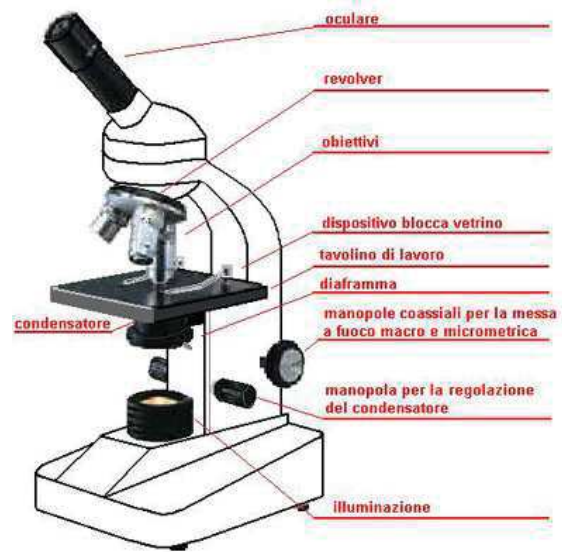


Figura 5 Microscopio ottico.

I raggi luminosi centrati sul campione dal condensatore sono poi raccolti dalle lenti dell'obiettivo del microscopio. Ora bisogna considerare due gruppi di raggi luminosi che entrano nell'obiettivo: quelli modificati dal campione e quelli non modificati. Il secondo gruppo consiste in un cono di luce proveniente dalle lenti del condensatore che passa direttamente nell'obiettivo e crea la luce di fondo del campo visivo. Il primo gruppo di raggi luminosi è quello che emana, in un certo senso, dai vari punti che compongono il campione. Dal campione i raggi luminosi sono messi a fuoco dall'obiettivo per formare un'immagine reale e ingrandita dell'oggetto all'interno della colonna del microscopio. L'immagine formata dall'obiettivo è usata poi come oggetto da un secondo sistema di lenti, la lente oculare, per ottenere l'immagine ingrandita virtuale. Un terzo sistema di lenti, localizzato nella parte anteriore dell'occhio, usa l'immagine virtuale prodotta dagli oculari come un oggetto per generare un'immagine reale sulla retina. Quando si ruota la manopola per la messa a fuoco del microscopio, la distanza relativa tra il campione e l'obiettivo varia, permettendo all'immagine finale di essere messa a fuoco sul piano della retina. L'ingrandimento totale ottenuto dal microscopio è il prodotto dell'ingrandimento dato dall'obiettivo moltiplicato per quello degli oculari¹.

3.1 Microscopio a contrasto di fase

Piccoli campioni non colorati, come le cellule viventi, possono essere molto difficili da distinguere con un microscopio a campo chiaro. Il microscopio a contrasto di fase aggira questa difficoltà, rendendo più visibili oggetti altamente trasparenti. La possibilità di vedere le diverse parti di uno stesso oggetto dipende dalla loro capacità di interagire con la luce in maniera diversa l'una dall'altra. Una caratteristica per cui gli organelli intracellulari differiscono è il loro indice di rifrazione. Gli organelli cellulari sono composti da diverse proporzioni di varie molecole: DNA, RNA, proteine, lipidi, carboidrati, sali e acqua. Regioni a composizione diversa dovrebbero avere differente indice di rifrazione. Normalmente, però, tali differenze non possono essere percepite dai nostri occhi. Il microscopio a contrasto di fase converte le differenze di indice di rifrazione in differenze di intensità (luminosità e oscurità relative), che sono quindi visibili all'occhio. Il principio di questa conversione si basa sulla capacità delle onde luminose di interagire l'una con l'altra, proprietà detta interferenza. Per come è costituito, il microscopio a contrasto di fase compie due funzioni che il microscopio a campo chiaro non è

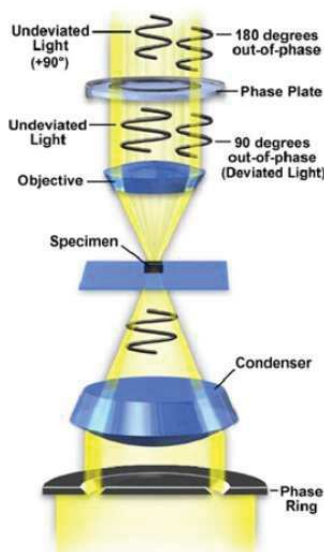


Figura 6 Microscopio a contrasto di fase.

in grado di svolgere: separa la luce diretta (la luce di fondo del campo) dalla luce rifratta dall'oggetto e rende questi due tipi di onde sfasati all'incirca di mezza lunghezza d'onda l'uno dall'altro cosicché essi possano interagire distruttivamente e creare cambiamenti di intensità. Presumiamo per un momento che tutte le parti dell'oggetto esaminato abbiano esattamente lo stesso indice di rifrazione del mezzo in cui sono sospese. In un caso del genere, tutta la luce proveniente dall'oggetto e la luce diretta sarebbero sfasate fra loro di mezza lunghezza d'onda e l'interferenza risulterebbe in una uniforme diminuzione di intensità. Ora, consideriamo un oggetto le cui diverse parti abbiano differenti indici di rifrazione, tutti maggiori del mezzo circostante. Le differenze di indice di rifrazione producono cambiamenti di fase dei raggi di luce che attraversano le diverse parti del campione. La luminosità o oscurità relativa di ogni parte dell'immagine riflette il modo in cui la luce proveniente da quella parte del campione interferisce con la luce diretta. Poiché le parti del campione hanno differente indice di rifrazione, le varie parti del campione appaiono nell'immagine con intensità variabile, fornendo il contrasto necessario a renderle visibili.

L'ottica per contrasto di fase è stata particolarmente utile nell'esame di componenti intracellulari di cellule vive ad una risoluzione relativamente alta. Per esempio, la mobilità dinamica dei mitocondri, i cromosomi mitotici e i vacuoli possono essere osservati e filmati. Il maggiore vantaggio derivato dall'invenzione del microscopio a contrasto di fase non è stata la scoperta di nuove strutture, ma il suo uso quotidiano, nella ricerca e nei laboratori didattici, per l'osservazione di cellule in un modo più dimostrativo².

3.2 Microscopia a fluorescenza

Certe molecole (dette fluorocromi) assorbono radiazioni di lunghezza d'onda ultravioletta, invisibile, e rilasciano parte dell'energia in forma di luce visibile con lunghezza d'onda maggiore, facendo sì che esse brillino su uno sfondo scuro. Data questa proprietà, la distribuzione di queste molecole all'interno di un tessuto può essere rilevata usando un microscopio ottico. In alcuni studi, il composto

fluorescente è iniettato in una cellula viva e il suo destino è seguito mentre la cellula svolge le sue normali attività. Per esempio, i fluorocromi possono essere usati per studiare le dimensioni di molecole in grado di passare tra le cellule, come indicatori di potenziali transmembrana, o come sonde per determinare la concentrazione di calcio libero nel citosol. In alternativa, il fluorocromo può essere legato in modo covalente (coniugato) a una particolare proteina, quale l'actina o la tubulina, e la proteina marcata con fluorescenti può essere iniettata in una cellula viva. Le proteine marcate partecipano alle normali attività della cellula, e la loro posizione può essere seguita al microscopio per rivelare le attività dinamiche alle quali le proteine stesse partecipano. In modo analogo, se anticorpi fluorescenti sono iniettati in una cellula, o aggiunti a una sezione di tessuto, si può determinare la posizione dell'antigene corrispondente. I fluorocromi possono essere usati anche in studi che utilizzano ibridazioni di DNA-RNA.

3.3 Microscopia a scansione confocale

Uno dei metodi più efficaci (e il più tedioso) nella raccolta di informazioni sulla struttura interna di un oggetto biologicamente interessante, come la disposizione spaziale dei vari tipi di cellule che compongono un organo, è quello di tagliare il blocco di tessuto in una serie di sezioni successive, chiamate sezioni seriali. Ogni sezione seriale può quindi essere esaminata

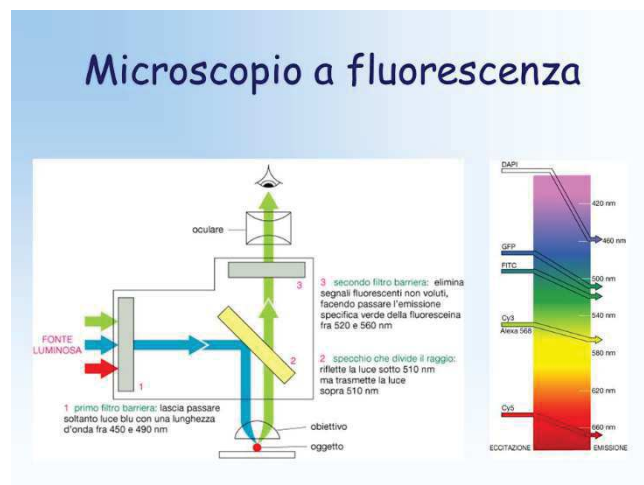


Figura 7 Microscopio a fluorescenza.

separatamente e l'organizzazione tridimensionale dell'intero oggetto può essere ricostruita sia nella memoria visiva, sia con l'aiuto di fotografie. Con lo sviluppo di sistemi elettronici e di produzione e di processamento delle immagini è diventato possibile esaminare "sezioni" di un oggetto senza doverlo tagliare con una lama. Il fatto che il campione contenga differenti livelli di fuoco riduce la possibilità di formare un'immagine nitida perché quelle parti del campione situate sopra e sotto il piano focale interferiscono con i raggi luminosi provenienti dalla parte che è sul piano focale. Negli ultimi dieci anni, è diventato disponibile un nuovo tipo di microscopio ottico che elimina questi problemi; il microscopio a scansione confocale. In questo tipo di microscopio, il campione è illuminato da un raggio laser focalizzato finemente che passa rapidamente attraverso il campione a una determinata profondità illuminando così soltanto un sottile piano all'interno dell'oggetto. Il microscopio è usato di solito con campioni colorati con fluorescenti e la luce emessa dalla sezione ottica illuminata è usata per formare un'immagine della sezione su uno schermo video. È evidente da questa serie che gli oggetti fuori dal piano focale hanno poco effetto sulla qualità dell'immagine di ogni singola sezione³.

CAPITOLO 4 – PROCEDURA ALGORITMO

Le cellule oggetto del nostro studio sono hter-hpne, cellule pre-tumorali del pancreas; hanno due oncosoppressori inattivati e un oncogene mutato.

Le immagini sono state ottenute attraverso l'utilizzo di un microscopio Nikon TI2 con ottica a contrasto di fase.

4.1 Descrizione delle immagini ottenute al microscopio

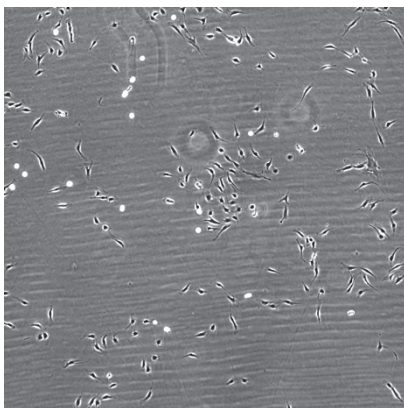


Figura 8 Cellule al microscopio.

Nelle immagini ottenute al microscopio ottico si possono subito osservare le cellule di nostro interesse, caratterizzate da un nucleo nero e da un citoplasma che lo circonda di colore grigio scuro. Queste non hanno una forma predefinita e immutabile bensì dipendente dal movimento: in posizione statica il nucleo tende ad assumere una

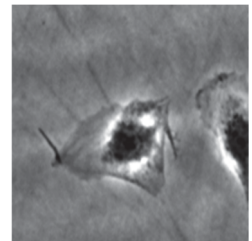


Figura 9 Cellula a riposo.

forma rotonda, in movimento la cellula si

allunga in direzione del suo moto; viceversa il citoplasma assume forme del tutto casuali indipendentemente dal moto. Lo sfondo su cui si muovono le cellule, ovvero la coltura cellulare, è di colore grigio, leggermente più chiaro

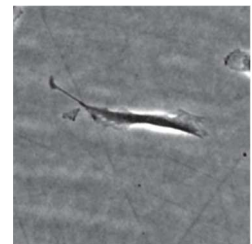


Figura 10 Cellula in movimento.

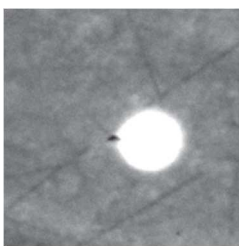


Figura 11 Cellula in corso di mitosi.

rispetto al citoplasma. Altro fattore da noi preso in considerazione è il ciclo cellulare composto da una fase di crescita cellulare e da una relativa alla divisione cellulare. Quest'ultima comporta, nella maggior parte dei casi, la genesi di due cellule figlie a partire dalla

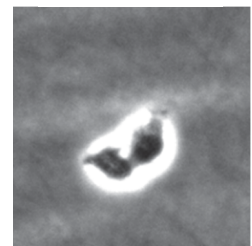


Figura 12 Cellula in tarda mitosi.

cellula preesistente. Di conseguenza non basterà seguire i

movimenti compiuti dalle cellule ma è necessario anche considerare la possibilità che queste effettuino la mitosi e quindi vadano incontro a duplicazione. Le cellule in attiva divisione cellulare possono essere facilmente individuate nel momento in cui la cromatina presente all'interno del nucleo, composta di proteine e DNA, è molto condensata perché in duplicazione; ciò fa sì che le cellule in mitosi appaiano bianche a seguito di



Figura 13 Cellula in mitosi precoce.

un cambiamento di fase della luce del microscopio e di forma rotondeggiante. Nonostante ciò può capitare che, in fase precoce o tardiva della mitosi, la cellula non sia un vero e proprio

cerchio bianco. In fase precoce si può intravedere parte della cellula madre, in fase tardiva parte delle due cellule figlie in fase di divisione. Un altro dato da considerare è che al microscopio ottico non risultano bianche solo le cellule in mitosi ma possono essere visualizzati degli aloni bianchi che circondano anche le cellule in fase di riposo.

L'ingresso principale del nostro algoritmo sono le immagini sopra descritte.

4.2 Elaborazione delle cellule

Il primo passaggio per andare a creare delle favorevoli condizioni di tracking consiste proprio nell'andare a modificare tali fotogrammi affinché vengano eliminati tutti gli ingressi interferenti. Quello che si vuole ottenere è un'immagine con sfondo nero in cui le cellule siano di colore bianco. Questa scelta è relativa al fatto che diverse funzioni di Matlab considerano i soggetti delle immagini di colore bianco.

In primis viene utilizzata la funzione 'imread' che va a leggere l'immagine di interesse per poi salvarla in una variabile. A questo punto si procede utilizzando la funzione 'im2bw' che converte l'immagine in una binaria (ovvero formata da 1 e 0 corrispondenti rispettivamente al bianco e al nero) a seconda della soglia scelta. Il valore scelto come soglia è 0,2. L'immagine risultante è caratterizzata da uno sfondo bianco e da cellule di colore nero. Dovendo visualizzare i soggetti della ricerca (ovvero le cellule) di colore bianco, viene utilizzato 'imcomplement'. Quest'ultima ha l'esclusivo scopo di invertire i numeri 1 con gli 0 e viceversa.

È stata riscontrata la presenza nello sfondo di punti bianchi, rimossi applicando "bwareaopen"; tutti gli oggetti composti da meno di X pixel sono eliminati. Considerata l'evenienza che alcune cellule in primo piano di colore bianco possano presentare dei puntini neri causa di interferenza con successive analisi, si utilizza 'imfill' che convertirà i pixel neri circondati da bianchi in quest'ultimo colore.



Figura 14 Immagine a seguito dell'utilizzo di 'im2bw'.



Figura 15 Immagine ottenuta utilizzando 'imcomplement'.



Figura 16 Immagine risultante l'utilizzo di 'bwareaopen'.

Nel momento in cui si visualizza le cellule del fotogramma, ci si può imbattere nella presenza di una cellula divisa in due parti. Utilizzando “strel” si viene a creare una linea verticale e una orizzontale, applicando “imdilate” si procede con la dilatazione dei bordi dei soggetti in questione. Con tali passaggi si ottengono degli ottimi fotogrammi di partenza per la nostra analisi.

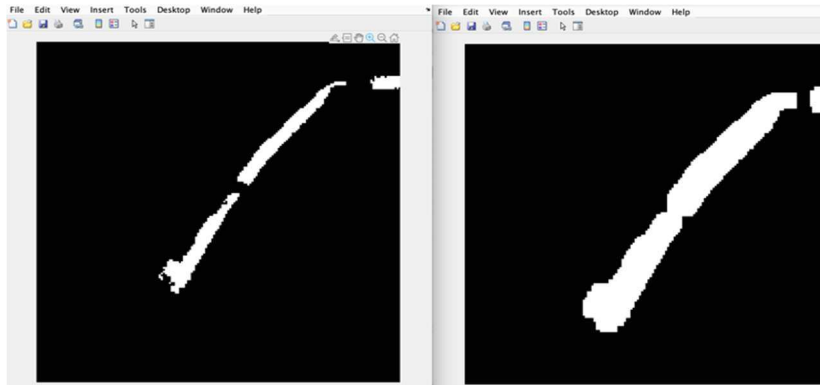


Figura 17 Prima e dopo l'utilizzo di 'imdilate'.

4.3 Centroidi

Per determinare la posizione di una cellula è necessario scegliere un punto di riferimento che verrà utilizzato nel corso di tutto l'algoritmo. Una soluzione è data dalla scelta del centroide, ovvero il centro geometrico di una figura. Si determina quindi il centroide di ogni cellula e successivamente lo si salva in una matrice. Il fotogramma in binario è rappresentato da una matrice di M colonne e N righe. Ogni casella di questa contiene 0 o 1 a seconda se il pixel nell'immagine sia nero o bianco. Nel caso in esame si possiede uno sfondo con molteplici 0 (nero) e delle regioni caratterizzate da numeri 1 (le cellule). Utilizzando 'bwlabel' si ottiene una matrice identica a quella rappresentante l'immagine, con la differenza che ogni cellula viene identificata con un numero differente; in questo modo è possibile distinguerle l'una dall'altra. Di seguito utilizzando 'regionprops'

Figura 18 Immagine visualizzata in binario.

Fields	Centroid
1	[15.1442,800.4599]
2	[30.7028,1.7074e+03]
3	[2.7374e+03,2.8011e+03]
4	[66.3354,1.6190e+03]
5	[93.3840,2.6603e+03]
6	[99.1736,916.6449]
7	[103.4954,2.3114e+03]
8	[121.7471,889.8725]
9	[118.7649,1.4844e+03]
10	[150.1477,2.6900e+03]
11	[150.2714,1.6293e+03]
12	[210.7566,911.2618]
13	[232.9224,363.3006]
14	[239.5017,468.8015]
15	[270.3416,2.8735e+03]
16	[323.9729,1.3839e+03]
17	[333.1405,442.0370]
18	[337.1448,3.0388e+03]
19	[362.9439,982.9154]
20	[363.1132,610.9178]
21	[376.7834,2.9641e+03]

Figura 19 Matrice dei centroidi ottenuta utilizzando 'regionprops'.

viene calcolato il centro di ogni regione le cui coordinate (x, y) vengono salvate su un array; i centri in esame sono proprio i centroidi delle cellule.

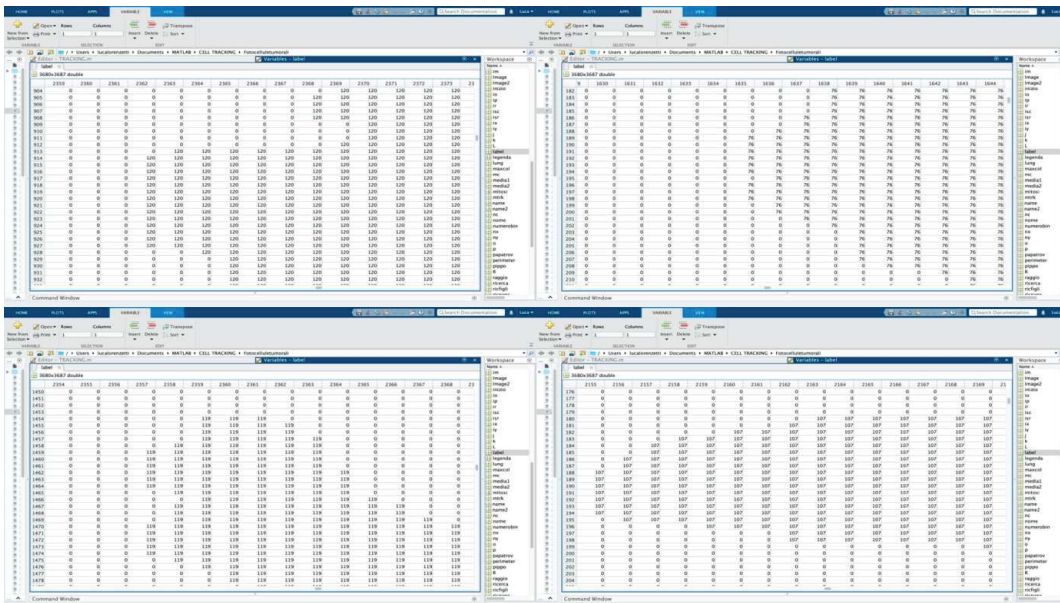


Figura 20 Matrice ottenuta attraverso 'bwlabel'.

4.4 Rilevazione dello spostamento cellulare

L'idea alla base del tracking si fonda sul fatto che le cellule, considerato che le immagini vengono catturate ogni 45 minuti, non compiono spostamenti rilevanti tra un fotogramma e l'altro. Perciò, per andare a tracciare il movimento, basta considerare che la sovrapposizione di due fotogrammi adiacenti fa sì che il centroide della cellula si possa trovare su quello successivo in un range massimo di circa 60 pixel. A questo punto, considerato un centroide di un fotogramma, viene calcolata la distanza di questo con ogni centroide del fotogramma successivo; sono state considerate solo le distanze sotto una determinata soglia scelta, nel nostro caso 60 pixel. La distanza minima individua quale centroide nel fotogramma successivo corrisponde alla stessa cellula considerata. Tale ragionamento viene ripetuto per ogni fotogramma e per ogni cellula utilizzando dei cicli 'for'.

La matrice che si ottiene è composta da tante colonne quante sono le cellule analizzate nel corso dello studio; si compone di tante righe quanti sono i fotogrammi analizzati. In dettaglio, in una stessa colonna ci sono le coordinate delle posizioni assunte dalla stessa cellula nel corso dei vari fotogrammi.

	5	6	7	8	9	10	11	12	13	14	15	16
1	...	[112.1041,1.3631e+03]	[152.084...	[282.2700,1.3323e+03]	[272.8451,540.1289]	[257.624...	[320.867...	[378.557...	[349.265...	[335.4414,2.8696e+03]	[364.302...	
2	...	[114.4927,1.3607e+03]	[182.406...	[295.5455,1.3421e+03]	[264.9630,526.2498]	[254.981...	[310.112...	[392.866...	[357.846...	[332.4768,2.8649e+03]	[369.373...	
3	...	[110.2671,1.3732e+03]	[162.177...	[302.1856,1.3492e+03]	[261.2318,517.4070]	[254.465...	[304.346...	[422.230...	[364.423...	[317.3462,2.8542e+03]	[363.170...	
4	...	[109.2775,1.3890e+03]	[140.274...	[303.9166,1.3549e+03]	[256.9159,507.8884]	[251.239...	[308.621...	[468.744...	[370.035...	[299.1559,2.8474e+03]	[361.376...	
5	...	[114.0632,1.4284e+03]	[124.468...	[312.7728,1.3648e+03]	[251.6669,498.6915]	[245.455...	[316.389...	[479.003...	[368.408...	[286.1340,2.8531e+03]	[359.246...	
6	...	[118.2103,1.4414e+03]	[115.094...	[313.8984,1.3697e+03]	[248.3294,492.2920]	[239.497...	[315.357...	[487.167...	[368.889...	[283.4354,2.8589e+03]	[349.691...	
7	...	[122.8137,1.4740e+03]	[113.630...	[319.4307,1.3761e+03]	[243.5454,479.7712]	[233.603...	[313.751...	[504.704...	[365.933...	[269.3804,2.8734e+03]	[336.647...	
8	...	[118.7649,1.4844e+03]	[121.747...	[323.9729,1.3839e+03]	[239.5017,468.8015]	[232.922...	[333.140...	[513.031...	[362.943...	[270.3416,2.8735e+03]	[337.144...	

Figura 21 Matrice rappresentante le coordinate dei centroidi delle cellule.

Precedentemente è stata utilizzata una funzione per dilatare i bordi delle cellule affinché non ci siano errori nella visualizzazione di una cellula divisa in più parti. Nonostante ciò, questa dilatazione deve essere esigua altrimenti l'eventuale presenza di gruppi di cellule molto vicine provocherebbe l'unione di due cellule differenti. Di conseguenza si è usato un altro metodo per ovviare a ciò; vediamo con un esempio concreto. Nel fotogramma i -esimo si ha la cellula A le cui coordinate sono nella colonna x . Nel fotogramma i -esimo+1 la cellula A appare spezzata in 2 parti. L'algoritmo cerca la cellula A nel fotogramma i -esimo+1 e associa la cellula A a una delle due parti più vicina al centroide della cellula A; l'altra parte verrà considerata come una nuova cellula e posizionata su una nuova colonna. Nel fotogramma i -esimo+2 la cellula A torna ad essere visualizzata come una cellula unica. L'algoritmo cerca il centroide più vicino a entrambe le parti sul fotogramma successivo e associa a entrambi un unico centroide relativo alla cellula A. Nella matrice contenente tutti i centroidi delle cellule ci sono, come risultato di tale errore, due colonne con gli stessi valori; utilizzando un ciclo 'for' vengono eliminate tali colonne.



Figura 22 Cellula vista nel corso di tre fotogrammi consecutivi.

4.5 Elaborazione delle cellule in fase mitotica

Nei fotogrammi presi in considerazione fino ad ora non si ha presenza di cellule in mitosi. Come effettuato inizialmente con le cellule, si applicano delle modifiche alle immagini affinché le mitosi risultino bianche su sfondo nero.

Come detto precedentemente, le mitosi risultano al microscopio già bianche, perciò sfruttando tale colore si può eliminare il resto. Le funzioni utilizzate e già descritte precedentemente sono:

- “imbinarize”, con una soglia alta così da lasciare solo ciò che è molto bianco;
- “bwareaopen”, per eliminare oggetti con aree piccole;



Figura 23 Immagine ottenuta applicando 'imbinarize'.

- “strel” e “imdilate”, per dilatare i bordi (nel caso in cui alcune mitosi possano sembrare divise in due);
- “imfill”, per rendere bianchi quei punti neri che possono apparire nelle cellule in mitosi;



Figura 24 Immagine ottenuta dopo l'utilizzo di 'bwareaopen'.

Nelle immagini risultanti sono ancora presenti gli aloni delle cellule. Per poter discriminare una mitosi è stata utilizzata la loro caratteristica di essere all'incirca rotonde.

Utilizzando “bwboundaries” si ottiene i punti perimetrali degli oggetti presenti nel fotogramma. Successivamente con “regionprops” sono state ricavate le aree e i centroidi dei vari oggetti.

Area	Centroid
1075	[95.5972,899.6893]
847	[111.1570,926.5773]
938	[102.4254,1.4903e+03]
1065	[133.5221,1.4779e+03]
920	[200.3489,923.2011]
1061	[221.6409,901.4166]
1739	[268.5877,2.8738e+03]
1016	[259.9675,486.8504]
1067	[316.0187,1.3955e+03]
2332	[336.7431,126.4528]
800	[350.5013,974.1050]
1015	[381.1576,979.9389]
945	[509.8296,933.6275]
1174	[513.3373,269.3961]
701	[528.5321,109.8459]
1108	[565.4702,118.0153]

Figura 25 Matrice ottenuta tramite 'regionprops'.

Un indicatore considerato per vedere la rotondità è l'indice di rotondità, ottenuto effettuando $4 \cdot \text{pigreco} \cdot \text{area} / \text{perimetro}^2$.

Calcolando quest'ultimo per ognuno degli oggetti presenti e scegliendo un valore minimo che deve possedere un oggetto per essere considerato mitosi è possibile eliminare tutti gli aloni. Una verifica ulteriore è effettuata con la funzione “raggioinscritto”: inscrive in ogni oggetto un cerchio di massimo raggio R, tutti gli oggetti che hanno un raggio minore di una determinata soglia vengono scartati. Viene creata una matrice per salvare tutti i centroidi delle mitosi.

Alcune cellule stazionano nel processo di mitosi più a lungo di altre, ne consegue che alcune mitosi durano più di un fotogramma.

L'obiettivo è quello di creare una matrice contenente informazioni sia sul fotogramma che sulle coordinate di inizio e di fine di ogni mitosi, il nome della cellula madre e quello delle cellule figlie (corrispondente al numero della colonna della matrice in cui vengono raggruppati tutti i centroidi delle cellule) e relative coordinate.

Dopo aver individuato una mitosi, l'algoritmo memorizza il fotogramma d'inizio, poi cerca la presenza di un centroide di una mitosi nel fotogramma successivo in un range di circa 30 pixel. La positività al centroide fa sì che l'algoritmo proceda con il controllo nel fotogramma successivo; in caso negativo memorizza l'ultimo fotogramma in cui è presente. Inoltre ad ogni mitosi è necessario associare una cellula madre e le cellule figlie. La cellula madre si trova nel fotogramma precedente a quello di inizio della mitosi ad una distanza scelta; nel caso in cui ci fossero più cellule intorno al centroide della mitosi, l'algoritmo sceglie quella più vicina.

Figura 26 Matrice contenente informazioni sulle mitosi.

inizio	icoord	fine	fcoord	papa	coordpapa	coordfigli	nomefig1	coordfiglio2
24	[773.5169,1.4633e+03]	24	[773.5169,1.463...	285	[786.2781,1.4518e+03]	[779.0782,...	594	[767.0611,1.4953e+03]
24	[805.7998,1.8859e+03]	25	[813.8451,1.897...	521	[819.2289,1.9340e+03]	[764.8853,...	639	[879.8257,1.9256e+03]
24	[1.8265e+03,1.9346e+03]	24	[1.8265e+03,1.9...	549	[1.8336e+03,1.9213e...	[1.8135e+0...	549	[1.8415e+03,1.9789e+...
24	[2.0208e+03,1.5815e+03]	24	[2.0208e+03,1.5...	410	[2.0092e+03,1.5752e...	[2.0148e+0...	600	[2.0312e+03,1.6760e+...
24	[2.2229e+03,1.5242e+03]	25	[2.2299e+03,1.5...	528	[2.2245e+03,1.5257e...	[2.2509e+0...	624	[2.1751e+03,1.5138e+...
24	[2.5726e+03,583.5420]	24	[2.5726e+03,583...	554	[2.5541e+03,629.8981]	[2.6028e+0...	605	[2.6625e+03,551.8079]

Per determinare le cellule figlie l'algoritmo analizza l'immagine successiva a quella della fine della mitosi e cerca due cellule che siano in un certo range di distanza dal centroide della mitosi. In varie occasioni nei fotogrammi possono esserci delle mitosi precoci o tardive, come spiegato inizialmente. L'algoritmo, nella determinazione della cellula madre, non è in grado di distinguere né la cellula in fase di riposo né quella in corso di mitosi, lo stesso vale anche per le cellule figlie. Non potendo attuare questa distinzione, l'algoritmo cerca sul fotogramma precedente o successivo a quello attuale, a seconda che sia rispettivamente alla ricerca della cellula madre o di quelle figlie. Il limite di fotogrammi da valutare è dettato da quanti minuti intercorrono tra l'acquisizione di un'immagine e un'altra.

4.6 Velocità e direzione delle cellule

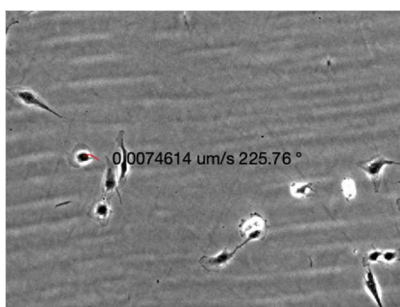


Figura 27 Visualizzazione della velocità e direzione di una cellula.

Di fondamentale importanza per lo scopo dello studio è la valutazione della velocità e della direzione di ogni cellula in ogni fotogramma. La velocità è data dal rapporto fra lo spazio e il tempo. Lo spazio è rappresentato dai pixel che una cellula percorre fra un fotogramma e quello successivo mentre il tempo è relativo ai secondi che intercorrono fra uno scatto e quello successivo, nel caso in questione 2700 secondi. L'unità di misura del parametro velocità è pixel su secondi. Attraverso un fattore di conversione è possibile trasformare la dimensione dei pixel in micrometri così da comprendere in maniera più intuitiva lo spostamento effettuato. La direzione viene ricavata con 'atan2d', ovvero l'inverso della tangente del vettore che congiunge le due posizioni occupate dalla cellula in due fotogrammi successivi, il cui risultato è un angolo sempre positivo fra 0° e 360°.

4.7 Interfaccia grafica (GUI)

Affinché tale algoritmo possa essere utilizzato anche da utenti che non conoscono Matlab, si è rivelato necessario progettare un'interfaccia utente facilmente approcciabile che potesse

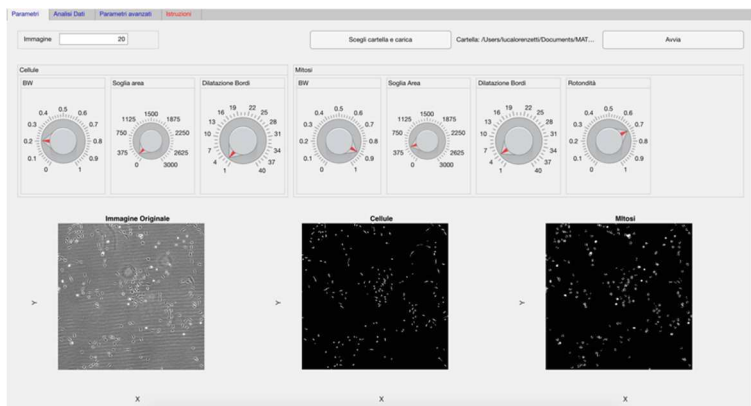


Figura 28 Interfaccia utente.

semplificare la scelta dei possibili parametri e la lettura dei dati forniti. Ciò può essere effettuato attraverso App Designer. L'output creato si compone di una prima sezione intitolata 'Parametri'. Questa offre la possibilità di modificare i vari tipi di parametri presenti nell'algoritmo

che si riferiscono alla visualizzazione delle cellule in fase di riposo o durante la divisione cellulare. Per quanto concerne i parametri delle cellule non coinvolte da mitosi, si può scegliere:

- La soglia del bianco e del nero, ovvero il parametro che separa ciò che è bianco da ciò che è nero così da mettere maggiormente in rilievo le cellule prese in considerazione;
- La soglia riguardante l'area sotto la quale nessun oggetto è considerabile cellula ma solo punti interferenti. Tutti gli oggetti con area inferiore ad essa vengono eliminati;
- La dilatazione dei bordi delle cellule.

Riguardo le cellule in fase mitosi, si può scegliere:

- La soglia del bianco e del nero;
- La soglia dell'area;
- La dilatazione dei bordi;
- L'indice di rotondità; più è alto più verranno considerate mitosi solo gli oggetti più rotondi.

Man mano che si cambiano i parametri vengono visualizzate le immagini con le modifiche effettuate in real-time.

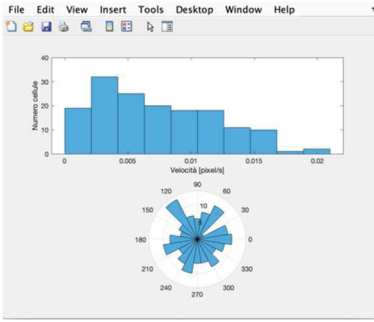


Figura 29 Istogramma rappresentativo delle velocità delle cellule.

Nella sezione 'Analisi dati' è possibile scegliere il fotogramma a seguito del quale vengono visualizzate il numero di mitosi e le cellule presenti in esso, un istogramma che mostra la distribuzione di velocità delle cellule e un altro che mostra visivamente il tasso di crescita delle cellule e delle mitosi nel corso dei vari fotogrammi. In questa sezione è disponibile una funzione per creare un video nel quale è mostrato tutto il percorso effettuato dalla cellula e nel caso in cui fosse presente una mitosi, anche quello delle cellule figlie.

L'output è anche fornito di una sezione apposita nella quale vengono fornite tutte le istruzioni per permettere l'utilizzo dell'algoritmo a mani non esperte.

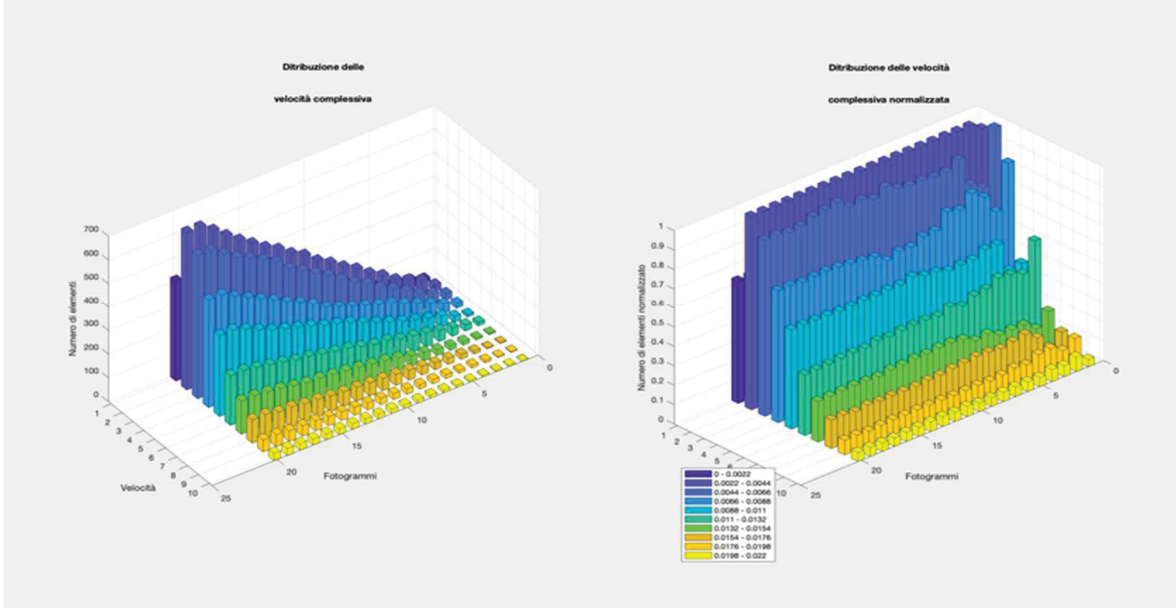


Figura 30 Istogramma in funzione del numero dei fotogrammi, delle cellule e delle loro velocità.

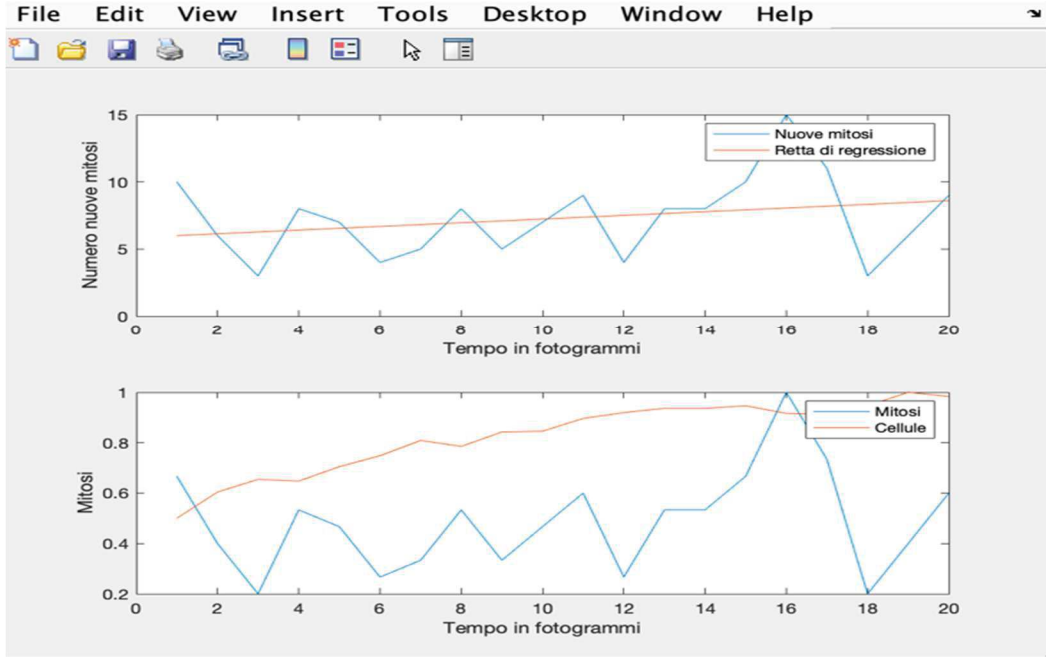


Figura 31 Grafico relativo all'andamento della crescita cellulare.

4.8 Codice algoritmo

```
classdef Algoritmotracking < matlab.apps.AppBase
```

```
    % Properties that correspond to app components
```

```
properties (Access = public)
```

```
    UIFigure          matlab.ui.Figure
    TabGroup          matlab.ui.container.TabGroup
    ParametriTab      matlab.ui.container.Tab
    GridLayout3       matlab.ui.container.GridLayout
    GridLayout4       matlab.ui.container.GridLayout
    Panel_4           matlab.ui.container.Panel
    ImmagineEditField_3Label  matlab.ui.control.Label
    ImmagineEditField      matlab.ui.control.NumericEditField
    AvviaButton        matlab.ui.control.Button
    ScegliscartellaecaricaButton  matlab.ui.control.Button
    CartellaselezionatanessunaLabel  matlab.ui.control.Label
    GridLayout5        matlab.ui.container.GridLayout
    CellulePanel        matlab.ui.container.Panel
    GridLayout7         matlab.ui.container.GridLayout
    BWPanel             matlab.ui.container.Panel
    GridLayout8         matlab.ui.container.GridLayout
    BWKnob_2           matlab.ui.control.Knob
    SogliaareaPanel     matlab.ui.container.Panel
    GridLayout9         matlab.ui.container.GridLayout
    SogliaareaKnob_2    matlab.ui.control.Knob
    DilatazioneBordiPanel  matlab.ui.container.Panel
    GridLayout10        matlab.ui.container.GridLayout
    DilatazionebordiKnob_2  matlab.ui.control.Knob
    MitosiPanel         matlab.ui.container.Panel
    GridLayout11        matlab.ui.container.GridLayout
    BWPanel_2           matlab.ui.container.Panel
    GridLayout12        matlab.ui.container.GridLayout
    BWKnob              matlab.ui.control.Knob
    SogliaAreaPanel     matlab.ui.container.Panel
    GridLayout12_2      matlab.ui.container.GridLayout
    SogliaareaKnob      matlab.ui.control.Knob
```

DilatazioneBordiPanel_2	matlab.ui.container.Panel
GridLayout12_3	matlab.ui.container.GridLayout
DilatazionebordiKnob	matlab.ui.control.Knob
RotonditPanel	matlab.ui.container.Panel
GridLayout12_4	matlab.ui.container.GridLayout
RotonditKnob	matlab.ui.control.Knob
GridLayout6	matlab.ui.container.GridLayout
UIAxes	matlab.ui.control.UIAxes
UIAxes3	matlab.ui.control.UIAxes
UIAxes2	matlab.ui.control.UIAxes
AnalisiDatiTab	matlab.ui.container.Tab
GridLayout14	matlab.ui.container.GridLayout
GridLayout15	matlab.ui.container.GridLayout
ANALISIPERFOTOGRAMMALabel	matlab.ui.control.Label
GridLayout16	matlab.ui.container.GridLayout
ImmagineEditField_2	matlab.ui.control.NumericEditField
NumeroimmagineLabel	matlab.ui.control.Label
NumerocellulepresentinelfotogrammaselezionatoLabel	matlab.ui.control.Label
NumerodimitosipresentinelfotogrammaselezionatoLabel	matlab.ui.control.Label
BottonepermostrarellistogrammaButton	matlab.ui.control.Button
GridLayout17	matlab.ui.container.GridLayout
ANALISICOMPLESSIVALabel	matlab.ui.control.Label
AndamentocelluleemitosiButton	matlab.ui.control.Button
IstogrammaButton	matlab.ui.control.Button
GridLayout18	matlab.ui.container.GridLayout
PERCORSOCOMPIUTODAUNACELLULALabel	matlab.ui.control.Label
GridLayout19	matlab.ui.container.GridLayout
LetteredellacolonnaexcelEditField	matlab.ui.control.EditField
ColonnaExcelLabel	matlab.ui.control.Label
PercorsodellacellulaButton	matlab.ui.control.Button
ParametriavanzatiTab	matlab.ui.container.Tab
GridLayout20	matlab.ui.container.GridLayout
NumerofotogrammiEditFieldLabel	matlab.ui.control.Label
NumerofotogrammiEditField	matlab.ui.control.NumericEditField
RicercaмитosiEditFieldLabel	matlab.ui.control.Label
RicercaмитosiEditField	matlab.ui.control.NumericEditField

```
RicercaCellulaPadreEditFieldLabel matlab.ui.control.Label
RicercaCellulaPadreEditField matlab.ui.control.NumericEditField
RicercaCelluleFiglieEditFieldLabel matlab.ui.control.Label
RicercaCelluleFiglieEditField matlab.ui.control.NumericEditField
DistanzaEditFieldLabel matlab.ui.control.Label
DistanzaEditField matlab.ui.control.NumericEditField
TempoEditFieldLabel matlab.ui.control.Label
TempoEditField matlab.ui.control.NumericEditField
IstruzioniTab matlab.ui.container.Tab
GridLayout13 matlab.ui.container.GridLayout
PremerePerAprireIlManualeDelleIstruzioniDeLlAlgoritmoButton matlab.ui.control.Button
```

```
end
```

```
properties (Access = private)
```

```
Property % Description
```

```
immagine = [];
```

```
immaginebw = [];
```

```
BW = [];
```

```
BW2 = [];
```

```
BWc = [];
```

```
BWc1 = [];
```

```
BWc2 = [];
```

```
BWc3 = [];
```

```
Immagine = [];
```

```
Vel = [];
```

```
Trk = [];
```

```
Mitosi = [];
```

```
Mtrk = [];
```

```
Lettera = [];
```

```
Ctrk = [];
```

```
Num = 0;
```

```
Ricerca = [5];
```

```
Ricpapa = [2];
```

```
Ricfigli = [2];
```

```
Distpapa = [50];
```

```
Tempo = [2700];
Numerofiles = [];
path = [];
scelta = 0;
end
```

```
% Callbacks that handle component events
methods (Access = private)
```

```
% Code that executes after component creation
function start(app)
drawnow;
app.UIFigure.WindowState = 'maximized';
end
```

```
% Value changed function: BWKnob
function bwmitosi(app, event)
value = app.BWKnob.Value;
immbw1 = imread(app.immagine);
app.immaginebw = imbinarize(immbw1,value);
%app.immaginebw = imresize(app.immaginebw,3);% bianco tutto quello che è sotto value
imshow(app.immaginebw,'Parent',app.UIAxes2);
end
```

```
% Value changed function: SogliaareaKnob
function Areamitosi(app, event)
value = app.SogliaareaKnob.Value;
value = round(value,0); % la soglia deve essere un numero intero
try
app.BW = bwareaopen(app.immaginebw,value); % elimina le aree sotto 283 che sono punti bianchi
imshow(app.BW,'Parent',app.UIAxes2);
catch ME
uiaalert(app.UIFigure, strcat(ME.message, 'Prima muovi BW'), 'Image Error');
end
```

```
end
```

```
% Value changed function: DilatazionebordiKnob
```

```
function Bordimitosi(app, event)
```

```
value = app.DilatazionebordiKnob.Value;
```

```
try
```

```
se90 = strel('line',value,90);
```

```
se0 = strel('line',value,0);
```

```
DA = imdilate(app.BW,[se90 se0]); % dilata i bordi
```

```
app.BW2 = imfill(DA,'holes'); % riempie i buchi
```

```
imshow(app.BW2,'Parent',app.UIAxes2);
```

```
catch ME
```

```
uialert(app.UIFigure, strcat(ME.message, 'Prima muovi 1 e 2'), 'Image Error');
```

```
end
```

```
end
```

```
% Value changed function: RotonditKnob
```

```
function rotmitosi(app, event)
```

```
value = app.RotonditKnob.Value;
```

```
try
```

```
[B,L] = bwboundaries(app.BW2,'noholes'); % in B coordinate dei punti perimetrali
```

```
AeC = regionprops(L,'Area','Centroid');
```

```
rot = value; % più è alto più prende i centri degli oggetti più rotondi,escludendo gli altri oggetti
```

```
imshow(app.BW2,'Parent',app.UIAxes2);
```

```
hold(app.UIAxes2,"on");
```

```
% calcola la rotondità di ogni chiazza nell'immagine delle mitosi
```

```
if length(B)>0
```

```
for k = 1 : length(B)
```

```
boundary = B{k}; % coordinate dei punti di ogni chiazza
```

```
% perimetro e aerea delle chiazza
```

```
delta_sq = diff(boundary).^2;
```

```
perimeter = sum(sqrt(sum(delta_sq,2)));
```

```
area = AeC(k).Area;
```

```
% rapporto tra area e perimetro (indice di rotondezza)
```

```
ir = 4*pi*area/perimeter^2;
```

```

% memorizzazione delle coordinate dei centri delle mitosi
if ir > rot
mc = AeC(k).Centroid; % contiene i centroidi delle mitosi
plot(app.UIAxes2,mc(1),mc(2),'ro');
end
end
end
hold(app.UIAxes2,"off");
catch ME
uialert(app.UIFigure,strcat(ME.message,'Prima muovi 1 e 2 e 3'),'Image Error');
end
end

```

```

% Value changed function: BWKnob_2

```

```

function bwcellule(app, event)
value = app.BWKnob_2.Value;
try
bw = imread(app.immagine);
bw = im2bw(bw,value); %facciamo diventare nero quello che è sopra la soglia 'value' e il resto bianco
app.BWc = imcomplement(bw); %inverte bianco e nero
imshow(app.BWc,'Parent',app.UIAxes3);
catch ME
uialert(app.UIFigure,strcat(ME.message,'Prima muovi 1 e 2 e 3'),'Image Error');
end
end

```

```

% Callback function

```

```

function sogliareacellule1(app, event)
value = app.Sogliaarea1Knob.Value;
value = round(value,0);
try
bw2 = bwareaopen(app.BWc,value); % elimina le aree sotto 'value'
app.BWc1 = imfill(bw2,'holes');
imshow(app.BWc1,'Parent',app.UIAxes3);
catch ME

```



```

uiaalert(app.UIFigure, strcat(ME.message, 'Prima muovi 1 e 2 e 3'), 'Image Error');
end
end

```

```

% Value changed function: SogliaareaKnob_2

```

```

function sogliaareacellule2(app, event)
value = app.SogliaareaKnob_2.Value;
value = round(value,0);
try
app.BWc2 = bwareaopen(app.BWc,value); % eliminiamo i punti bianchi con area sotto 'value'
imshow(app.BWc2, 'Parent', app.UIAxes3);
catch ME
uiaalert(app.UIFigure, strcat(ME.message, 'Prima muovi 1 e 2 e 3'), 'Image Error');
end
end

```

```

% Value changed function: DilatazionebordiKnob_2

```

```

function Dilatbordicellule(app, event)
value = app.DilatazionebordiKnob_2.Value;
try
se90 = strel('line',value,90);
se0 = strel('line',value,0);
DA = imdilate(app.BWc2,[se90 se0]); % dilata i bordi
app.BWc3 = imfill(DA, 'holes'); % riempie i buchi
imshow(app.BWc3, 'Parent', app.UIAxes3);
catch ME
uiaalert(app.UIFigure, strcat(ME.message, 'Prima muovi 1 e 2'), 'Image Error');
end
end

```

```

% Button pushed function: AvviaButton

```

```

function avvia(app, event)
ricerca = app.Ricerca; % limite di fotogrammi da considerare dopo aver trovato una mitosi per vedere se
dura più di 1 fotogramma

```

```

ricpapa = app.Ricpapa; % ricerca il padre fino a 2 fotogrammi dopo
ricfigli = app.Ricfigli; % ricerca i figli fino a 2 fotogrammi dopo
distpapa = app.Distpapa; % area di ricerca del padre
tempo = app.Tempo; % tempo fra un fotogramma e l'altro
maxcol = 0;
trk = zeros(100,100);
mtrk = zeros(100,100);
%% POSIZIONA NELLA MATRICE 'TRK' LE COORDINATE DEI CENTROIDI RILVEATI NEL SUSSEGUIRSI DEI FO-
TOGRAMMI
% NELLE COLONNE PARI CI SONO LE Y,NELLE DISPARI CI SONO LE X, UN CENTROIDE E' IDENTIFICATO DALLA
PRIMA COLONNA
% E DALLA SECONDA E COSI' VIA
f1 = uifigure;
d1 = uiprogressdlg(f1,"Title","Elaborazione dati in corso",...
'Message','Caricamento immagini in corso');
%fa scegliere la cartella se l'operatore si è dimenticato
if app.scelta == 0
selpath = uigetdir;
cd(selpath);
app.CartellaselezionatanessunaLabel.Text = strcat("Cartella: ",selpath);
end
conserva = 0;
flist = dir('*.*');
%elimina tutte le cartelle da questa lista
flist([flist.isdir]==1)= [];
temptemp = zeros(numel(flist),1);
for cellidx = 1 : numel(flist)-1 %numel is a lot safer than length
if strcmp(flist(cellidx).name, '.DS_Store')
flist(cellidx) = [];
end
end
flist = natsortfiles({flist.name});
for i = 1 : numel(flist) %numel is a lot safer than length
tempim = imread(flist{i});
name=strcat(int2str(i),'.png'); %il file finale sarà una png
if size(tempim,3) > 1

```

```

if i == 1
%domanda conservazione file se l'immagine ha più
%layer
answer = questdlg("Il programma per ridurre l'occupazione cancella le immagini originali, si desidera con-
servarle? ", ...
'Conservare immagini originali?', ...
'Sì','No','No');
% Handle response
switch answer
case 'Sì'
disp('Sono salvate nella cartella')
conserva = 1;
case 'No'
conserva = 0;
end
%crea la cartella con le immagini conservate
if conserva == 1
mkdir('old');
end
end
ingr = tempim(:, :, 1); %legge solo il primo 'layer' (su 3)
try
imwrite(ingr, name);
if conserva == 1
movefile(flist{i}, 'old')
else
delete(flist{i});
end
catch
end
flist = dir('*. *');
%elimina tutte le cartelle da questa lista
flist([flist.isdir]==1)= [];
temptemp = zeros(numel(flist),1);
for cellidx = 1 : numel(flist)-1 %numel is a lot safer than length
if strcmp(flist(cellidx).name, '.DS_Store')

```

```

flist(cellidx) = [];
end
end
flist = natsortfiles({flist.name});
end
end
%controlli sul numero di foto da analizzare
%se non è stato impostato alcun valore si analizzano tutte le
%foto
if app.Num == 0
app.Num = numel(flist);
end
%se il numero di foto da analizzare è maggiore del numero di
%foto presenti, le si analizza tutte
if app.Num > numel(flist)
app.Num = numel(flist);
end
for i = 1 : app.Num
name = flist{i};
%% SOVRAPPOSIZIONE DELL'IMMAGINE OTTENUTA ORIMA CON EDGE E POI CON LA DILATAZIONE DEI
BORDI
% Rilevazione cellule
Image = imread(name);
% [~,rot] = edge(Image,'sobel'); % trova i contorni della cellula
% fudgeFactor = 0.5 ;
% BWs = edge(Image,'sobel',rot * fudgeFactor);
%% CAMBIO SOGLIA
bw = im2bw(Image,app.BWKnob_2.Value); %facciamo diventare nero quello che è sopra la soglia 0.3 e il
resto bianco
bw = imcomplement(bw); %inverte bianco e nero
bw3 = imfill(bw,'holes');
bw5 = bwareaopen(bw3,app.SogliaareaKnob_2.Value); % eliminiamo i punti bianchi con area sotto 120
se90 = strel('line',app.DilatazionebordiKnob_2.Value,90);
se0 = strel('line',app.DilatazionebordiKnob_2.Value,0);
bwA = imdilate(bw5,[se90 se0]); % dilata i bordi
d1.Value = .1;

```

```

%% TROVA I CENTROIDI DELLE CELLULE E LI PLOTTA SULL'IMMAGINE
label = bwlabel(bwA); % ogni regione ha un numero diverso
stat = regionprops(label,'centroid'); % calcola il centro di ogni regione e li mette in 'stat'
T = numel(stat);
if T*2 > maxcol
maxcol = T*2;
end
assign = zeros(T,1); % matrice con tante righe quanti sono i centroidi
d1.Value = .2;
%% Mettiamo su un vettore trk le coordinate dei centroidi delle cellule
if(i==1)
for nc = 0 : numel(stat)-1 % numero cellule va da 1 a numero centroidi
trk(i,1+nc*2) = stat(nc+1).Centroid(1);
trk(i,2+nc*2) = stat(nc+1).Centroid(2);
end
ldv = maxcol/2; % ldv contiene il numero di centroidi del fotogramma i-1
else
for i_trk = 0 : ldv-1
id_min = 0;
Z = 999999999999999.9;
for i_stat = 0 : numel(stat)-1 % indice che scorre tutti i centroidi dell'i-esimo fotogramma
if trk(i-1,1+i_trk*2) < 0 % non calcolo la distanza tra -999999 e un centroide
continue;
end
lung = lng(trk(i-1,1+i_trk*2),trk(i-1,2+i_trk*2),stat(i_stat+1).Centroid(1),stat(i_stat+1).Centroid(2));
if lung<Z
Z = lung;
id_min = i_stat+1;
end
end
if Z > 60 % se la distanza più piccola tra un centroide dell'i-esimo fotogramma e quello prima è maggiore 60
vuol dire che la cellula del fotogramma prima non c'è sul fotogramma dopo,soglia di incolonnamento di
stesse cellule in fotogrammi successivi
trk(i,2*i_trk+1) = -999999999 ;
trk(i,2*i_trk+2) = -999999999 ;
else

```

```

trk(i,2*i_trk+1) = stat(id_min).Centroid(1);
trk(i,2*i_trk+2) = stat(id_min).Centroid(2);
assign(id_min) = 1;
end
end
end
if i>1
for p = 1 : T
if assign(p) == 0 % i centroidi nuovi li aggiunge in fondo a trk
trk(i,maxcol+1) = stat(p).Centroid(1);
trk(i,maxcol+2) = stat(p).Centroid(2);
maxcol = maxcol+2;
end
end
end
ldv = maxcol/2; % ldv contiene il numero di centroidi del fotogramma i-1
end
d1.Value = .3;
d1.Message = 'Estrazione delle mitosi';
%% MEMORIZZA I CENTROIDI DELLE MITOSI
for im = 1 : app.Num
name2 = flist{im};
Image2 = imread(name2);
j = 0;
% Mitosi
BW = imbinarize(Image2,app.BWKnob.Value); % bianco tutto ciò che è sotto 0.9
BW2 = bwareaopen(BW,app.SogliaareaKnob.Value); % elimina le aree sotto 283 che sono punti bianchi
se90 = strel('line',app.DilatazionebordiKnob.Value,90);
se0 = strel('line',app.DilatazionebordiKnob.Value,0);
bib = imdilate(BW2,[se90 se0]); % dilata i bordi
BW3 = imfill(bib,'holes'); % riempie i buchi
% Elimina i punti non corrispondenti alla mitosi
[B,L] = bwboundaries(BW3,'noholes'); % in B coordinate dei punti perimetrali
AeC = regionprops(L,'Area','Centroid');
rot = app.RotonditKnob.Value; % più è alto più prende i centri degli oggetti più rotondi,escludendo gli altri
oggetti

```

```

raggio = zeros(length(B),5);
% calcola la rotondità di ogni chiazza nell'immagine delle mitosi
if length(B) > 0
for k = 1 : length(B)
boundary = B{k}; % coordinate dei punti di ogni chiazza
% perimetro e aerea delle chiazza
delta_sq = diff(boundary).^2;
perimeter = sum(sqrt(sum(delta_sq,2)));
area = AeC(k).Area;
% rapporto tra area e perimetro (indice di rotondezza)
ir = 4*pi*area/perimeter^2;
mc = AeC(k).Centroid; % è impostata come la matrice 'trk', contiene i centroidi delle mitosi
% memorizzazione delle coordinate dei centri delle mitosi
if ir > rot
j = j+1;
mtrk(im,2*j-1) = mc(1);
mtrk(im,2*j) = mc(2);
[R nx ny] = raggioinscritto(B{k});
raggio(k,1) = R;
raggio(k,3) = mc(1);
raggio(k,4) = mc(2);
raggio(k,5) = im;
else
[R nx ny] = raggioinscritto(B{k});
raggio(k,2)=R;
raggio(k,3) = mc(1);
raggio(k,4) = mc(2);
raggio(k,5) = im;
end
end
valori = raggio(:,1);
valori = valori(valori>0);
media1=mean(valori);
valori = raggio(:,2);
valori = valori(valori>0);
media2=mean(valori);

```

```

%%
for i = 1:size(raggio,1)
soglia = mean([media1;media2]);
if raggio(i,2) > soglia
j = j+1;
mtrk(im,2*j-1) = raggio(i,3);
mtrk(im,2*j) = raggio(i,4);
end
end
%%
end
end
d1.Value = .4;
d1.Message = 'Elaborazione delle cellule';
%% RAGGRUPPIAMO LE COORDINATE DI UN CENTROIDE(X,Y) IN UNA SINGOLA CELLA
ctrk = zeros(size(trk,1),size(trk,2)/2);
isr = 1;
isc = 2;
ctrk = num2cell(ctrk);
for isr = 1 : size(trk,1)
for isc = 2 : 2 : size(trk,2)
t1 = trk(isr,isc-1);
t2 = trk(isr,isc);
temp = [t1 t2];
ctrk{isr,isc/2} = temp;
end
end
d1.Value = .5;
%% ELIMINIAMO DA CTRK(MATRICE CHE CONTIENE TUTTI I CENTROIDI DELLE CELLULE),ALCUNE COLONNE
CHE RIPORTANO LE STESSE COORDINATE
%% DOVUTE AL NON RICONOSCIMENTO DI UNA CELLULA SINGOLA MA LE VEDE COME 2 SEPARATE
cloni =[] ;
for a = 1 : size(ctrk,2) % colonne
for b = 1 : size(ctrk,1) % righe
if ctrk{b,a}(1) < 0 % se incontra -999999 cambia colonna
break;

```



```

end
if ctrk{b,a}{1} == 0 % se incontra uno 0 passa alla riga successiva
continue;
end
for t = 1 : size(ctrk,2)
if ctrk{b,a}{1} == ctrk{b,t}{1} && t > a
if ~any(ismember(cloni,t)) % se in clone non abbiamo già salvato la colonna di un clone la scrivo
cloni = [cloni t]; % vettore-riga contenente le colonne di ctrk che contengono le coordinate dei cloni
end
end
end
end
end
end

cloni = sort(cloni,'descend');
for a = 1 : size(cloni,2)
ctrk(:,cloni(a)) = [];
end
d1.Value = .7;

%% PROBLEMA MITOSI CHE DURANO PIU' FOTOGRAMMI, E MITOSI PARZIALI (non si vede nè mitosi nè
cellula)
copiamtrk = mtrk;
mitosi = [];
ix = 2; % righe di mtrk
iy = 1; % colonne di mtrk
ip = 1;
c = 1;
d = 1;
cricerca=ricerca;

% se una mitosi dura più di un fotogramma? creiamo una matrice 'mitosi' in
% cui ci sono i dettagli di dove inizia e finisce ogni mitosi
for ix = 1 : app.Num-1 % scorre le righe di mtrk
if ix+ricerca <= app.Num % non cerca mitosi su fotogrammi dopo la fine dei fotogrammi
for iy = 1 : 2 : size(mtrk,2) % scorre le colonne dei centroidi delle mitosi
if mtrk(ix,iy) ~= 0
s.inizio = ix;
s.icoord = [mtrk(ix,iy) mtrk(ix,iy+1)];

```

```

trovato = 0;
for c = 1 : ricerca
for d = 1 : 2 : size(mtrk,2)
if mtrk(ix,iy) == 0
continue;
end
lung = lng(mtrk(ix,iy),mtrk(ix,iy+1),mtrk(ix+c,d),mtrk(ix+c,d+1));
if lung<30 % mitosi che durano più fotogrammi,cerca mitosi su fotogramma successivo con distanza minore
a 30
s.fine = ix+c;
s.fcoord = [mtrk(ix+c,d) mtrk(ix+c,d+1)];
mtrk(ix+c,d) = 0;
mtrk(ix+c,d+1) = 0;
trovato = 1;
end
end
end
if trovato == 0
s.fine = s.inizio;
s.fcoord = s.icoord;
end
mitosi = [mitosi;s];
end
end
else
for iy = 1 : 2 : size(mtrk,2) % scorre le colonne dei centroidi delle mitosi
if mtrk(ix,iy) ~= 0
s.inizio = ix;
s.icoord = [mtrk(ix,iy) mtrk(ix,iy+1)];
trovato = 0;
cricerca = app.Num-ix;
for c = 1 : cricerca
for d = 1 : 2 : size(mtrk,2)
if mtrk(ix,iy) == 0
continue;
end

```

```

lung = lng(mtrk(ix,iy),mtrk(ix,iy+1),mtrk(ix+c,d),mtrk(ix+c,d+1));
if lung<30 % mitosi che durano più fotogrammi,cerca mitosi su fotogramma successivo con distanza minore
a 30
s.fine = ix+c;
s.fcoord = [mtrk(ix+c,d) mtrk(ix+c,d+1)];
mtrk(ix+c,d) = 0;
mtrk(ix+c,d+1) = 0;
trovato = 1;
end
end
end
if trovato == 0
s.fine = s.inizio;
s.fcoord = s.icoord;
end
mitosi = [mitosi;s];
end
end
end
end
d1.Value = .9;
%% TROVA I PAPA'
ix = 1;
io = 1;
for ix = 1 : size(mitosi,1)
inizio = mitosi(ix).inizio;
coordinate = mitosi(ix).icoord;
papatrov = 0;
for f = 1 : ricpapa
for io = 1 : size(ctrk,2)
if (inizio-f) > 0
if ctrk{inizio-f,io}{1} < 0 % se incontra -999999 cambia colonna
continue;
end
lung = lng(ctrk{inizio-f,io}{1},ctrk{inizio-f,io}{2},coordinate(1,1),coordinate(1,2));
if lung < distpapa

```

```

mitosi(ix).papa = io;
mitosi(ix).coordpapa = ctrk{inizio-f,io};
papatrov = 1;
end
end
end
if papatrov == 1 % se trova il padre non cerca sul altri fotogrammi
break;
end
end
end
%% TROVA I FIGLI
ix = 1;
io = 1;
coordinate = [];
for ix = 1 : size(mitosi,1)
fine = mitosi(ix).fine;
coordinate = mitosi(ix).fcoord;
fig1 = [0];
fig2 = [0];
for g = 1 : ricfigli
distm = 99999;
distM = 99999;
if fig1 == 0 || fig2 == 0
for io = 1 : size(ctrk,2) % indice che scorre tutti i centroidi delle cellule dell'i-esimo+1 fotogramma
if fine+g <= app.Num
if ctrk{fine+g,io}(1) < 0 % se incontra -999999 cambia colonna
continue;
end
lung = lng(ctrk{fine+g,io}(1),ctrk{fine+g,io}(2),coordinate(1,1),coordinate(1,2)); % calcolo la distanza fra
centroide mitosi e centroidi del fotogramma seguente la mitosi
if lung < 100 % soglia di plausibilità
if lung < distM % assegno a fig2 il secondo più piccolo (più grande di fig1)
fig2 = io;
distM = lung;
end

```

```

if lung < distm % assegno a fig1 il più piccolo
fig2 = fig1;
distM = distm;
fig1 = io;
distm = lung;
end
end
end

nome = mitosi(ix).papa;
if fig1 ~= 0
mitosi(ix).coordfiglio1 = ctrk{fine+g,fig1};
mitosi(ix).nomefig1 = fig1;
end
if fig2 ~= 0
mitosi(ix).coordfiglio2 = ctrk{fine+g,fig2};
mitosi(ix).nomefig2 = fig2;
end
end
end
end
end

%% CALCOLO VELOCITA' E DIREZIONE DI OGNI CELLULA
ix = 1;
iy = 1;
tempo = 2700; % 45 min
vel = zeros(app.Num,100);
vel = num2cell(vel);
for ix = 1 : app.Num-1
for iy = 1 : size(ctrk,2)
if ctrk{ix,iy}{1} > 0 && ctrk{ix+1,iy}{1} > 0
[velocita,angolo] = velEdir(ctrk{ix,iy}{1},ctrk{ix,iy}{2},ctrk{ix+1,iy}{1},ctrk{ix+1,iy}{2},tempo);
vel{ix,iy} = [velocita,angolo];
end
end
end
for x = 1 : size(vel,1)

```

```

for y = 1 : size(vel,2)
if size(vel{x,y}) == 1
vel{x,y} = [0,0];
end
end
end

app.Vel = vel;
app.Trk = trk;
app.Mitosi = mitosi;
app.Mtrk = mtrk;
app.Ctrk = ctrk;
d1.Value = 1;
close(f1);
mkdir('risultati');
cd('risultati');
writecell(vel,'Velocità&direzione.xlsx');
cd('..');
end

% Value changed function: ImmagineEditField
function Fotogramma(app, event)
if app.scelta == 0
scegliscartella(app,event);
end

value = app.ImmagineEditField.Value;
f1 = uifigure;
d1 = uiprogresdlg(f1, "Title", "Elaborazione dati in corso", ...
'Message', 'Caricamento immagine in corso');
listing = dir('*.*');
flist = [];
for cellidx = 1 : numel(listing) %numel is a lot safer than length
if listing(cellidx).isdir == 0
flist = [flist; listing(cellidx)];
end
end
temptemp = zeros(numel(flist),1);

```

```

for cellidx = 1 : numel(flist)-1 %numel is a lot safer than length
if strcmp(flist(cellidx).name, '.DS_Store')
flist(cellidx) = [];
end
end
flist = natsortfiles({flist.name});
%controlla se quell'immagine esiste effettivamente
%se ci sono 4 foto e l'operatore chiede la quinta il programma
%da un errore e chiede all'operatore di ripetere l'operazione
if value <= numel(flist)
%configurazione con valori di default per evitare e crash e iniziare le variabili
app.immagine = flist{value};
immbw1 = imread(app.immagine);
imshow(immbw1,'Parent',app.UIAxes);
immbw1 = imread(app.immagine);
app.immaginebw = imbinarize(immbw1,0.9); % bianco tutto quello che è sotto value
imshow(app.immaginebw,'Parent',app.UIAxes2);
try
app.BW = bwareaopen(app.immaginebw,120); % elimina le aree sotto 283 che sono punti bianchi
imshow(app.BW,'Parent',app.UIAxes2);
catch ME
uialet(app.UIFigure, strcat(ME.message, 'Prima muovi BW'), 'Image Error');
end
try
se90 = strel('line',2,90);
se0 = strel('line',2,0);
DA = imdilate(app.BW,[se90 se0]); % dilata i bordi
app.BW2 = imfill(DA,'holes'); % riempie i buchi
imshow(app.BW2,'Parent',app.UIAxes2);
catch ME
uialet(app.UIFigure, strcat(ME.message, 'Prima muovi 1 e 2'), 'Image Error');
end
value = app.RotonditKnob.Value;
d1.Value = .5
try
[B,L] = bwboundaries(app.BW2,'noholes'); % in B coordinate dei punti perimetrali

```

```

AeC = regionprops(L,'Area','Centroid');
rot = 1; % più è alto più prende i centri degli oggetti più rotondi,escludendo gli altri oggetti
imshow(app.BW2,'Parent',app.UIAxes2);
hold(app.UIAxes2,"on");
% calcola la rotondità di ogni chiazza nell'immagine delle mitosi
if length(B)>0
for k = 1 : length(B)
boundary = B{k}; % coordinate dei punti di ogni chiazza
% perimetro e aerea delle chiazza
delta_sq = diff(boundary).^2;
perimeter = sum(sqrt(sum(delta_sq,2)));
area = AeC(k).Area;
% rapporto tra area e perimetro (indice di rotondezza)
ir = 4*pi*area/perimeter^2;
% memorizzazione delle coordinate dei centri delle mitosi
if ir > rot
mc = AeC(k).Centroid; % contiene i centroidi delle mitosi
plot(app.UIAxes2,mc(1),mc(2),'ro');
end
end
end
hold(app.UIAxes2,"off");
catch ME
uiaert(app.UIFigure,strcat(ME.message,'Prima muovi errore 3'),'Image Error');
end
try
bw = imread(app.immagine);
bw = im2bw(bw,0.2); %facciamo diventare nero quello che è sopra la soglia 'value' e il resto bianco
app.BWc = imcomplement(bw); %inverte bianco e nero
imshow(app.BWc,'Parent',app.UIAxes3);
catch ME
uiaert(app.UIFigure,strcat(ME.message,'errore 4'),'Image Error');
end
try
bw2 = bwareaopen(app.BWc,120); % elimina le aree sotto 'value'
app.BWc1 = imfill(bw2,'holes');

```



```

imshow(app.BWc1,'Parent',app.UIAxes3);
catch ME
    uialert(app.UIFigure, strcat(ME.message, 'Prima errore 5'), 'Image Error');
end
try
    app.BWc2 = bwareaopen(app.BWc1,120); % eliminiamo i punti bianchi con area sotto 'value'
    imshow(app.BWc2,'Parent',app.UIAxes3);
catch ME
    uialert(app.UIFigure, strcat(ME.message, 'Prima errore 6'), 'Image Error');
end
try
    se90 = strel('line',2,90);
    se0 = strel('line',2,0);
    DA = imdilate(app.BWc2,[se90 se0]); % dilata i bordi
    app.BWc3 = imfill(DA,'holes'); % riempie i buchi
    imshow(app.BWc3,'Parent',app.UIAxes3);
catch ME
    uialert(app.UIFigure, strcat(ME.message, 'Prima errore 7'), 'Image Error');
end
else
end
d1.Value = 1;
close(f1);
end

% Button pushed function:
% BottonepermostrarellistogrammaButton
function Istogramma(app, event)
    velo = [];
    for i = 1 : size(app.Vel,2)
        if app.Vel{app.ImagineEditField_2.Value,i} > 0
            velo = [velo,app.Vel{app.Immag,i}(1)];
        end
    end
    figure;
    subplot(2,1,1);

```

```

histogram(velo,10);
xlabel("Velocità [pixel/s]");
ylabel("Numero cellule");
conta=0;
for i = 1:size(app.Trk,2)
if app.Trk(app.ImmagineEditField_2.Value,i)>0
conta=conta+1;
end
end
conta = conta/2;
app.NumerocellulepresentinelfotogrammaselezionatoLabel.Text = ['Numero di cellule presenti nel foto-
gramma selezionato : ',num2str(conta)];
x = 0;
for i = 1 : size(app.Mitosi)
if app.ImmagineEditField_2.Value >= app.Mitosi(i).inizio && app.ImmagineEditField_2.Value <= app.Mi-
tosi(i).fine
x = x+1;
end
end
app.NumerodimitosipresentinelfotogrammaselezionatoLabel.Text = ['Numero di mitosi presenti nel foto-
gramma selezionato : ',num2str(x)];
drawnow;
i=app.Immag;
temprow = [];
for o = 1 : size(app.Vel,2)
if size(app.Vel{i,o},2) > 0
if app.Vel{i,o}(1) > 0
temprow = [temprow app.Vel{i,o}(2)];
end
end
end
subplot(2,1,2);
polarhistogram(temprow,20);
end

% Value changed function: ImmagineEditField_2

```

```

function Fotogramma2(app, event)
app.Immag = app.ImmagineEditField_2.Value;
end

% Button pushed function: IstogrammaButton
function Istogramma2(app, event)
vel = app.Vel;
temprow = [];
numerobin = 10;
datahist = zeros(size(vel,1),numerobin);
datahistnorm = zeros(size(vel,1),numerobin);
for i = 1 : size(vel,1)
for o = 1 : size(vel,2)
if size(vel{i,o},1) > 0
if vel{i,o}(1) > 0
temprow = [temprow vel{i,o}(1)];
end
end
end
[tempbin,edges] = histcounts(temprow,numerobin);
datahist(i,:) = tempbin;
datahistnorm(i,:) = tempbin/max(tempbin);
end
legenda = cell(1,1);
temp1 = [];
for i= 1:numerobin
temp = strcat(num2str(edges(i)), " - ", num2str(edges(i+1)));
if i == 1
legenda{1} = temp;
else
legenda{end+1} = temp;
end
end
figure
subplot(1,2,1)
bar3(datahist,0.5);

```

```

ylabel("Fotogrammi");
xlabel("Velocità");
zlabel("Numero di elementi");
title(["Distribuzione delle" newline "velocità complessiva"]);
subplot(1,2,2)
bar3(datahistnorm,0.5);
ylabel("Fotogrammi");
xlabel("Velocità");
zlabel("Numero di elementi normalizzato");
title(["Distribuzione delle velocità" newline "complessiva normalizzata"]);
legend(legenda,'Location','southwest')
end

```

```

% Button pushed function: AndamentocelluleemitosiButton

```

```

function andamento(app, event)
%funziona solo dopo aver eseguito il tracking
contamitosi=zeros(app.Num,1);
contacellule=zeros(app.Num,1);
x=[1:app.Num]';
for i=1:size(app.Mitosi)
init=app.Mitosi(i).inizio;
contamitosi(init) = contamitosi(init)+1;
end
%conta le nuove mitosi nell'ultimo fotogramma
contaultimo=0;
for i=1:size(app.Mtrk,2)
if app.Mtrk(app.Num,i) >0
contaultimo=contaultimo+1;
end
end
contamitosi(app.Num)=contaultimo/2;
%plotting
subplot(2,1,1);
plot(x,contamitosi);
xlabel("Tempo in fotogrammi");
ylabel("Numero nuove mitosi");

```

```

%retta di regressione
p = polyfit(x,contamitosi,1);
y1 = polyval(p,[1:app.Num]);
hold on
plot(x,y1);
hold off

%conta cellule nella foto
for i = 1:app.Num
conta=0;
for o = 1:size(app.Trk,2)
if app.Trk(i,o)>0
conta=conta+1;
end
end
contacellule(i)=conta/2;
end

legend('Nuove mitosi','Retta di regressione');
subplot(2,1,2);
contamitosi=contamitosi/max(contamitosi); %valore normalizzato
plot(x,contamitosi);
contacellule=contacellule/max(contacellule); %valore normalizzato
hold on
plot(x,contacellule);
hold off
xlabel("Tempo in fotogrammi");
ylabel("Mitosi");
legend('Mitosi','Cellule');
end

% Button pushed function: PercorsodellacellulaButton
function percorso(app, event)
cellula=cellfun(@(x) (sum((double(x)-64).*26.^(length(x)-1:-1:0))),...
upper({app.Lettera}));
if cellula ~= 1
cellula = fix(cellula/2);
end

```

```

colonna = cellula;
prima = 0;
i = 1;
plotx = [];
ploty = [];
for x = 1 : size(app.Ctrk,1)
if app.Ctrk{x,colonna}(1) > 0
if prima == 0
inizio = x;
prima = 1;
end
name = strcat(int2str(x),'.png');
image = imread(name);
figure1 = figure;
for l = inizio : x
plotx = [plotx; app.Ctrk{x,colonna}(1)];
end
for l = inizio : x
ploty = [ploty; app.Ctrk{x,colonna}(2)];
end
imshow(image);
hold on;
plot(plotx,ploty,'r');
if x > 1
try
temp = app.Vel{x,colonna};
testo = num2str(temp(1))+ " um/s " +num2str(temp(2))+ " °";
text (app.Ctrk{x,(colonna)}(1)+100,app.Ctrk{x,(colonna)}(2),testo);
catch ME
end
end
hold off;
mkdir('video');
cd('video');
saveas(figure1, strcat("v",int2str(i),'.png'));
cd('..');

```

```

i = i+1;
close;
end
if app.Ctrk{x,colonna}(1) < 0
break;
end
end
%% AGGIUMGE AL VIDEO I FOTOGRAMMI DELLE MITOSI
finemit = 0;
inizmit = 0;
for f = 1 : length(app.Mitosi)
if app.Mitosi(f).inizio > x
break;
end
if ~isempty(app.Mitosi(f).coordpapa)
if app.Mitosi(f).inizio == x && app.Mitosi(f).coordpapa(1) == app.Ctrk{x-1,colonna}(1)
figlio1 = app.Mitosi(f).nomefig1; % mi salvo i nomi delle cellule figlie
figlio2 = app.Mitosi(f).nomefig2;
inizmit = app.Mitosi(f).inizio;
finemit = app.Mitosi(f).fine;
break;
end
end
end
if inizmit ~= 0
for g = 1 : (finemit-inizmit)+1
name = strcat(int2str(inizmit-1+g),'.png');
image = imread(name);
figure1 = figure;
imshow(image);
hold on;
plot(plotx,ploty,'r');
hold off;
cd('video');
saveas(figure1, strcat("v",int2str(i),'.png'));
cd('..');

```

```

i = i+1;
close;
end
% PLOTTO IL PERCORSO DEI FIGLI DELLA MITOSI
prima = 0;
prima1 = 0;
plotx1 = [];
ploty1 = [];
plotx2 = [];
ploty2 = [];
for x = 1 : size(app.Ctrk,1)
if isempty(figlio1) && isempty(figlio2)
break;
end
if ~isempty(figlio1)
if app.Ctrk{x,figlio1}(1) > 0
if prima == 0
inizio = x;
prima = 1;
end
for l = inizio : x
plotx1 = [plotx1; app.Ctrk{x,figlio1}(1)];
end
for l = inizio : x
ploty1 = [ploty1; app.Ctrk{x,figlio1}(2)];
end
end
end
if ~isempty(figlio2)
if app.Ctrk{x,figlio2}(2) > 0
if prima1 == 0
inizio1 = x;
prima1 = 1;
end
for l = inizio : x
plotx2 = [plotx2; app.Ctrk{x,figlio2}(1)];

```



```

end
for l = inizio : x
ploty2 = [ploty2; app.Ctrk{x,figlio2}{2}];
end
end
end
a = 0;
b = 0;
if ~isempty(figlio1)
if app.Ctrk{x,figlio1}{1} > 0
a = 1;
end
end
if ~isempty(figlio2)
if app.Ctrk{x,figlio2}{1} > 0
b = 1;
end
end
if a || b
name = strcat(int2str(x),'.png');
image = imread(name);
figure1 = figure;
imshow(image);
hold on;
plot(plotx,ploty,'r',plotx1,ploty1,'g',plotx2,ploty2,'y');
if x > 1
if ~isempty(figlio1)
try
temp = app.Vel{x,figlio1};
testo = num2str(temp(1))+ " um/s " +num2str(temp(2))+ " °";
text (app.Ctrk{x,(figlio1)}{1}+200,app.Ctrk{x,(figlio1)}{2},testo);
catch ME
end
end
if ~isempty(figlio2)
try

```

```

temp = app.Vel{x,figlio2};
testo = num2str(temp(1))+" um/s " +num2str(temp(2))+" °";
text (app.Ctrk{x,(figlio2)}{1)+200,app.Ctrk{x,(figlio2)}{2},testo);
catch ME
end
end
end
hold off;
cd('video');
saveas(figure1, strcat("v",int2str(i),'.png'));
cd('..');
i = i+1;
close;
end
end
end
%% Creare il video
cd('video');
video = VideoWriter('yourvideo.avi'); %create the video object
video.FrameRate = 1;
open(video); %open the file for writing
for ii=1:i-1 %where N is the number of images
I = imread(strcat("v",int2str(ii),'.png'));
writeVideo(video,I); %write the image to file
end
close(video); %close the file
implay('yourvideo.avi');
cd('..');
end

% Value changed function: LetteredellacolonnaexcelEditField
function lettera(app, event)
app.Lettera = app.LetteredellacolonnaexcelEditField.Value;
end

```

```
% Value changed function: NumerofotogrammiEditField
function Numerofotogrammi(app, event)
app.Num = app.NumerofotogrammiEditField.Value;
end
```

```
% Value changed function: RicercamitosiEditField
function Ricercamitosi(app, event)
app.Ricerca = app.RicercamitosiEditField.Value;
end
```

```
% Value changed function: RicercacellulapadreEditField
function Ricercacellulapadre(app, event)
app.Ricpapa = app.RicercacellulapadreEditField.Value;
end
```

```
% Value changed function: RicercacellulefiglieEditField
function Ricercacellulefiglie(app, event)
app.Ricfigli = app.RicercacellulefiglieEditField.Value;
end
```

```
% Value changed function: DistanzaEditField
function Ricercadistanza(app, event)
app.Distpapa = app.DistanzaEditField.Value;
end
```

```
% Value changed function: TempoEditField
function Tempofoto(app, event)
app.Tempo = app.TempoEditField.Value;
end
```

```
% Button pushed function:
% Premereperaprireilmanualedele istruzionidellalgoritmoButton
function Istruzioni(app, event)
```

```

open("Istruzioni.pdf");
end

% Button pushed function: ScegliscartellaecaricaButton
function scegliscartella(app, event)
attuale = pwd;
app.path = uigetdir(attuale, 'Selezione la cartella con le foto');
if ~strcmp(app.path, '')
cd(app.path);
app.scelta = 1;
app.CartellaselezionatanessunaLabel.Text = strcat("Cartella: ", app.path );
else
fig = uifigure;
uiaalert(fig, 'La cartella selezionata non è valida', 'Cartella non valida');
end
end
end

% Component initialization
methods (Access = private)

% Create UIFigure and components
function createComponents(app)

% Create UIFigure and hide until all components are created
app.UIFigure = uifigure('Visible', 'off');
app.UIFigure.Position = [100 100 1137 723];
app.UIFigure.Name = 'UI Figure';

% Create TabGroup
app.TabGroup = uitabgroup(app.UIFigure);
app.TabGroup.Position = [1 0 1277 724];

```

```
% Create ParametriTab
```

```
app.ParametriTab = uitab(app.TabGroup);  
app.ParametriTab.Title = 'Parametri';  
app.ParametriTab.ForegroundColor = [0 0 1];
```

```
% Create GridLayout3
```

```
app.GridLayout3 = uigridlayout(app.ParametriTab);  
app.GridLayout3.ColumnWidth = {'1x'};  
app.GridLayout3.RowHeight = {'1x', '5x', '7x'};
```

```
% Create GridLayout4
```

```
app.GridLayout4 = uigridlayout(app.GridLayout3);  
app.GridLayout4.ColumnWidth = {'1x', '1x', '1x', '1x', '1x'};  
app.GridLayout4.RowHeight = {'1x'};  
app.GridLayout4.Layout.Row = 1;  
app.GridLayout4.Layout.Column = 1;
```

```
% Create Panel_4
```

```
app.Panel_4 = uipanel(app.GridLayout4);  
app.Panel_4.Layout.Row = 1;  
app.Panel_4.Layout.Column = 1;
```

```
% Create ImmagineEditField_3Label
```

```
app.ImmagineEditField_3Label = uilabel(app.Panel_4);  
app.ImmagineEditField_3Label.HorizontalAlignment = 'right';  
app.ImmagineEditField_3Label.Position = [7 4 58 22];  
app.ImmagineEditField_3Label.Text = 'Immagine';
```

```
% Create ImmagineEditField
```

```
app.ImmagineEditField = uieditfield(app.Panel_4, 'numeric');  
app.ImmagineEditField.ValueChangedFcn = createCallbackFcn(app, @Fotogramma, true);  
app.ImmagineEditField.Position = [80 4 100 22];
```

% Create AvviaButton

```
app.AvviaButton = uibutton(app.GridLayout4, 'push');
app.AvviaButton.ButtonPushedFcn = createCallbackFcn(app, @avvia, true);
app.AvviaButton.Layout.Row = 1;
app.AvviaButton.Layout.Column = 5;
app.AvviaButton.Text = {'Avvia'; ''};
```

% Create ScegliscartellaecaricaButton

```
app.ScegliscartellaecaricaButton = uibutton(app.GridLayout4, 'push');
app.ScegliscartellaecaricaButton.ButtonPushedFcn = createCallbackFcn(app, @scegliscartella, true);
app.ScegliscartellaecaricaButton.Layout.Row = 1;
app.ScegliscartellaecaricaButton.Layout.Column = 3;
app.ScegliscartellaecaricaButton.Text = 'Scegli cartella e carica';
```

% Create CartellaselezionatanessunaLabel

```
app.CartellaselezionatanessunaLabel = uilabel(app.GridLayout4);
app.CartellaselezionatanessunaLabel.Layout.Row = 1;
app.CartellaselezionatanessunaLabel.Layout.Column = 4;
app.CartellaselezionatanessunaLabel.Text = 'Cartella selezionata: nessuna';
```

% Create GridLayout5

```
app.GridLayout5 = uigrdatagrid(app.GridLayout3);
app.GridLayout5.ColumnWidth = {'3x', '5x'};
app.GridLayout5.RowHeight = {'1x'};
app.GridLayout5.Layout.Row = 2;
app.GridLayout5.Layout.Column = 1;
```

% Create CellulePanel

```
app.CellulePanel = uipanel(app.GridLayout5);
app.CellulePanel.Title = 'Cellule';
app.CellulePanel.Layout.Row = 1;
app.CellulePanel.Layout.Column = 1;
```

```

% Create GridLayout7
app.GridLayout7 = uigridlayout(app.CellulePanel);
app.GridLayout7.ColumnWidth = {'1x', '1x', '1x'};
app.GridLayout7.RowHeight = {'1x'};
app.GridLayout7.Padding = [5 5 5 5];

% Create BWPanel
app.BWPanel = uipanel(app.GridLayout7);
app.BWPanel.Title = 'BW';
app.BWPanel.Layout.Row = 1;
app.BWPanel.Layout.Column = 1;

% Create GridLayout8
app.GridLayout8 = uigridlayout(app.BWPanel);
app.GridLayout8.ColumnWidth = {'1x'};
app.GridLayout8.RowHeight = {'1x'};
app.GridLayout8.Padding = [0 0 0 0];

% Create BWKnob_2
app.BWKnob_2 = uiknob(app.GridLayout8, 'continuous');
app.BWKnob_2.Limits = [0 1];
app.BWKnob_2.ValueChangedFcn = createCallbackFcn(app, @bwcellule, true);
app.BWKnob_2.Layout.Row = 1;
app.BWKnob_2.Layout.Column = 1;
app.BWKnob_2.Value = 0.2;

% Create SogliaareaPanel
app.SogliaareaPanel = uipanel(app.GridLayout7);
app.SogliaareaPanel.Title = 'Soglia area';
app.SogliaareaPanel.Layout.Row = 1;
app.SogliaareaPanel.Layout.Column = 2;

% Create GridLayout9

```

```

app.GridLayout9 = uigradlayout(app.SogliaareaPanel);
app.GridLayout9.ColumnWidth = {'1x'};
app.GridLayout9.RowHeight = {'1x'};
app.GridLayout9.Padding = [0 0 0 0];

% Create SogliaareaKnob_2
app.SogliaareaKnob_2 = uiknob(app.GridLayout9, 'continuous');
app.SogliaareaKnob_2.Limits = [0 3000];
app.SogliaareaKnob_2.ValueChangedFcn = createCallbackFcn(app, @sogliaareacellule2, true);
app.SogliaareaKnob_2.Layout.Row = 1;
app.SogliaareaKnob_2.Layout.Column = 1;
app.SogliaareaKnob_2.Value = 120;

% Create DilatazioneBordiPanel
app.DilatazioneBordiPanel = uipanel(app.GridLayout7);
app.DilatazioneBordiPanel.Title = 'Dilatazione Bordi';
app.DilatazioneBordiPanel.Layout.Row = 1;
app.DilatazioneBordiPanel.Layout.Column = 3;

% Create GridLayout10
app.GridLayout10 = uigradlayout(app.DilatazioneBordiPanel);
app.GridLayout10.ColumnWidth = {'1x'};
app.GridLayout10.RowHeight = {'1x'};
app.GridLayout10.Padding = [0 0 0 0];

% Create DilatazionebordiKnob_2
app.DilatazionebordiKnob_2 = uiknob(app.GridLayout10, 'continuous');
app.DilatazionebordiKnob_2.Limits = [1 40];
app.DilatazionebordiKnob_2.ValueChangedFcn = createCallbackFcn(app, @Dilatbordicellule, true);
app.DilatazionebordiKnob_2.Layout.Row = 1;
app.DilatazionebordiKnob_2.Layout.Column = 1;
app.DilatazionebordiKnob_2.Value = 2;

```



```

% Create MitosiPanel
app.MitosiPanel = uipanel(app.GridLayout5);
app.MitosiPanel.Title = 'Mitosi';
app.MitosiPanel.Layout.Row = 1;
app.MitosiPanel.Layout.Column = 2;

% Create GridLayout11
app.GridLayout11 = uigridlayout(app.MitosiPanel);
app.GridLayout11.ColumnWidth = {'1x', '1x', '1x', '1x', '1x'};
app.GridLayout11.RowHeight = {'1x'};
app.GridLayout11.Padding = [5 5 5 5];

% Create BWPanel_2
app.BWPanel_2 = uipanel(app.GridLayout11);
app.BWPanel_2.Title = 'BW';
app.BWPanel_2.Layout.Row = 1;
app.BWPanel_2.Layout.Column = 1;

% Create GridLayout12
app.GridLayout12 = uigridlayout(app.BWPanel_2);
app.GridLayout12.ColumnWidth = {'1x'};
app.GridLayout12.RowHeight = {'1x'};
app.GridLayout12.Padding = [0 0 0 0];

% Create BWKnob
app.BWKnob = uiknob(app.GridLayout12, 'continuous');
app.BWKnob.Limits = [0 1];
app.BWKnob.ValueChangedFcn = createCallbackFcn(app, @bwmitosi, true);
app.BWKnob.Layout.Row = 1;
app.BWKnob.Layout.Column = 1;
app.BWKnob.Value = 0.9;

% Create SogliaAreaPanel

```

```

app.SogliaAreaPanel = uipanel(app.GridLayout11);
app.SogliaAreaPanel.Title = 'Soglia Area';
app.SogliaAreaPanel.Layout.Row = 1;
app.SogliaAreaPanel.Layout.Column = 2;

% Create GridLayout12_2
app.GridLayout12_2 = uigrayout(app.SogliaAreaPanel);
app.GridLayout12_2.ColumnWidth = {'1x'};
app.GridLayout12_2.RowHeight = {'1x'};
app.GridLayout12_2.Padding = [0 0 0 0];

% Create SogliaareaKnob
app.SogliaareaKnob = uiknob(app.GridLayout12_2, 'continuous');
app.SogliaareaKnob.Limits = [0 3000];
app.SogliaareaKnob.ValueChangedFcn = createCallbackFcn(app, @Areamitosi, true);
app.SogliaareaKnob.Layout.Row = 1;
app.SogliaareaKnob.Layout.Column = 1;
app.SogliaareaKnob.Value = 400;

% Create DilatazioneBordiPanel_2
app.DilatazioneBordiPanel_2 = uipanel(app.GridLayout11);
app.DilatazioneBordiPanel_2.Title = 'Dilatazione Bordi';
app.DilatazioneBordiPanel_2.Layout.Row = 1;
app.DilatazioneBordiPanel_2.Layout.Column = 3;

% Create GridLayout12_3
app.GridLayout12_3 = uigrayout(app.DilatazioneBordiPanel_2);
app.GridLayout12_3.ColumnWidth = {'1x'};
app.GridLayout12_3.RowHeight = {'1x'};
app.GridLayout12_3.Padding = [0 0 0 0];

% Create DilatazionebordiKnob
app.DilatazionebordiKnob = uiknob(app.GridLayout12_3, 'continuous');

```

```
app.DilatazionebordiKnob.Limits = [1 40];
app.DilatazionebordiKnob.ValueChangedFcn = createCallbackFcn(app, @Bordimitosi, true);
app.DilatazionebordiKnob.Layout.Row = 1;
app.DilatazionebordiKnob.Layout.Column = 1;
app.DilatazionebordiKnob.Value = 4;
```

```
% Create RotonditPanel
```

```
app.RotonditPanel = uipanel(app.GridLayout11);
app.RotonditPanel.Title = 'Rotondità';
app.RotonditPanel.Layout.Row = 1;
app.RotonditPanel.Layout.Column = 4;
```

```
% Create GridLayout12_4
```

```
app.GridLayout12_4 = uigridlayout(app.RotonditPanel);
app.GridLayout12_4.ColumnWidth = {'1x'};
app.GridLayout12_4.RowHeight = {'1x'};
app.GridLayout12_4.Padding = [0 0 0 0];
```

```
% Create RotonditKnob
```

```
app.RotonditKnob = uiknob(app.GridLayout12_4, 'continuous');
app.RotonditKnob.Limits = [0 1];
app.RotonditKnob.ValueChangedFcn = createCallbackFcn(app, @rotmitosi, true);
app.RotonditKnob.Layout.Row = 1;
app.RotonditKnob.Layout.Column = 1;
app.RotonditKnob.Value = 0.7;
```

```
% Create GridLayout6
```

```
app.GridLayout6 = uigridlayout(app.GridLayout3);
app.GridLayout6.ColumnWidth = {'1x', '1x', '1x'};
app.GridLayout6.RowHeight = {'1x'};
app.GridLayout6.Layout.Row = 3;
app.GridLayout6.Layout.Column = 1;
```

```

% Create UIAxes
app.UIAxes = uiaxes(app.GridLayout6);
title(app.UIAxes, 'Immagine Originale')
xlabel(app.UIAxes, 'X')
ylabel(app.UIAxes, 'Y')
app.UIAxes.PlotBoxAspectRatio = [1.28448275862069 1 1];
app.UIAxes.Layout.Row = 1;
app.UIAxes.Layout.Column = 1;

% Create UIAxes3
app.UIAxes3 = uiaxes(app.GridLayout6);
title(app.UIAxes3, 'Cellule')
xlabel(app.UIAxes3, 'X')
ylabel(app.UIAxes3, 'Y')
app.UIAxes3.PlotBoxAspectRatio = [1.28448275862069 1 1];
app.UIAxes3.Layout.Row = 1;
app.UIAxes3.Layout.Column = 2;

% Create UIAxes2
app.UIAxes2 = uiaxes(app.GridLayout6);
title(app.UIAxes2, 'Mitosi')
xlabel(app.UIAxes2, 'X')
ylabel(app.UIAxes2, 'Y')
app.UIAxes2.PlotBoxAspectRatio = [1.28448275862069 1 1];
app.UIAxes2.Layout.Row = 1;
app.UIAxes2.Layout.Column = 3;

% Create AnalisiDatiTab
app.AnalisiDatiTab = uitab(app.TabGroup);
app.AnalisiDatiTab.Title = 'Analisi Dati';
app.AnalisiDatiTab.ForegroundColor = [0 0 1];

% Create GridLayout14
app.GridLayout14 = uigridlayout(app.AnalisiDatiTab);

```

```
app.GridLayout14.ColumnWidth = {'1x'};
app.GridLayout14.RowHeight = {'1x', '1x', '1x'};
```

```
% Create GridLayout15
```

```
app.GridLayout15 = uigridlayout(app.GridLayout14);
app.GridLayout15.ColumnWidth = {'1x', '1x', '1x'};
app.GridLayout15.RowHeight = {'1x', '1x', '1x', '1x'};
app.GridLayout15.Layout.Row = 1;
app.GridLayout15.Layout.Column = 1;
```

```
% Create ANALISIPERFOTOGRAMMALabel
```

```
app.ANALISIPERFOTOGRAMMALabel = uilabel(app.GridLayout15);
app.ANALISIPERFOTOGRAMMALabel.HorizontalAlignment = 'center';
app.ANALISIPERFOTOGRAMMALabel.FontSize = 22;
app.ANALISIPERFOTOGRAMMALabel.FontWeight = 'bold';
app.ANALISIPERFOTOGRAMMALabel.Layout.Row = 1;
app.ANALISIPERFOTOGRAMMALabel.Layout.Column = 2;
app.ANALISIPERFOTOGRAMMALabel.Text = 'ANALISI PER FOTOGRAMMA';
```

```
% Create GridLayout16
```

```
app.GridLayout16 = uigridlayout(app.GridLayout15);
app.GridLayout16.RowHeight = {'1x'};
app.GridLayout16.Padding = [5 5 5 5];
app.GridLayout16.Layout.Row = 2;
app.GridLayout16.Layout.Column = 1;
```

```
% Create ImmagineEditField_2
```

```
app.ImagineEditField_2 = uieditfield(app.GridLayout16, 'numeric');
app.ImagineEditField_2.ValueChangedFcn = createCallbackFcn(app, @Fotogramma2, true);
app.ImagineEditField_2.Layout.Row = 1;
app.ImagineEditField_2.Layout.Column = 2;
```

```
% Create NumeroimmagineLabel
```

```
app.NumeroimmagineLabel = uilabel(app.GridLayout16);
app.NumeroimmagineLabel.Layout.Row = 1;
app.NumeroimmagineLabel.Layout.Column = 1;
app.NumeroimmagineLabel.Text = 'Numero immagine';
```

```
% Create Numero cellule presentinelfotogramma selezionato Label
```

```
app.Numero cellule presentinelfotogramma selezionato Label = uilabel(app.GridLayout15);
app.Numero cellule presentinelfotogramma selezionato Label.Layout.Row = 3;
app.Numero cellule presentinelfotogramma selezionato Label.Layout.Column = 1;
app.Numero cellule presentinelfotogramma selezionato Label.Text = 'Numero cellule presenti nel foto-
gramma selezionato';
```

```
% Create Numero di mitosi presentinelfotogramma selezionato Label
```

```
app.Numero di mitosi presentinelfotogramma selezionato Label = uilabel(app.GridLayout15);
app.Numero di mitosi presentinelfotogramma selezionato Label.Layout.Row = 4;
app.Numero di mitosi presentinelfotogramma selezionato Label.Layout.Column = 1;
app.Numero di mitosi presentinelfotogramma selezionato Label.Text = 'Numero di mitosi presenti nel foto-
gramma selezionato';
```

```
% Create Bottone per mostrare l'Istogramma Button
```

```
app.Bottone per mostrare l'Istogramma Button = uibutton(app.GridLayout15, 'push');
app.Bottone per mostrare l'Istogramma Button.ButtonPushedFcn = createCallbackFcn(app, @Istogramma,
true);
app.Bottone per mostrare l'Istogramma Button.Layout.Row = 3;
app.Bottone per mostrare l'Istogramma Button.Layout.Column = 2;
app.Bottone per mostrare l'Istogramma Button.Text = 'Bottone per mostrare l''Istogramma';
```

```
% Create GridLayout17
```

```
app.GridLayout17 = uigridlayout(app.GridLayout14);
app.GridLayout17.ColumnWidth = {'1x', '1x', '2x', '1x', '1x'};
app.GridLayout17.Layout.Row = 2;
app.GridLayout17.Layout.Column = 1;
```

```
% Create ANALISICOMPLESSIVALabel
```

```
app.ANALISICOMPLESSIVALabel = uilabel(app.GridLayout17);  
app.ANALISICOMPLESSIVALabel.HorizontalAlignment = 'center';  
app.ANALISICOMPLESSIVALabel.FontSize = 25;  
app.ANALISICOMPLESSIVALabel.FontWeight = 'bold';  
app.ANALISICOMPLESSIVALabel.Layout.Row = 1;  
app.ANALISICOMPLESSIVALabel.Layout.Column = 3;  
app.ANALISICOMPLESSIVALabel.Text = 'ANALISI COMPLESSIVA';
```

```
% Create AndamentocelluleemitosiButton
```

```
app.AndamentocelluleemitosiButton = uibutton(app.GridLayout17, 'push');  
app.AndamentocelluleemitosiButton.ButtonPushedFcn = createCallbackFcn(app, @andamento, true);  
app.AndamentocelluleemitosiButton.Layout.Row = 2;  
app.AndamentocelluleemitosiButton.Layout.Column = 2;  
app.AndamentocelluleemitosiButton.Text = 'Andamento cellule e mitosi';
```

```
% Create IstogrammaButton
```

```
app.IstogrammaButton = uibutton(app.GridLayout17, 'push');  
app.IstogrammaButton.ButtonPushedFcn = createCallbackFcn(app, @Istogramma2, true);  
app.IstogrammaButton.Layout.Row = 2;  
app.IstogrammaButton.Layout.Column = 4;  
app.IstogrammaButton.Text = 'Istogramma';
```

```
% Create GridLayout18
```

```
app.GridLayout18 = uigriddlayout(app.GridLayout14);  
app.GridLayout18.ColumnWidth = {'1x', '1x', '1x'};  
app.GridLayout18.Layout.Row = 3;  
app.GridLayout18.Layout.Column = 1;
```

```
% Create PERCORSOCOMPIUTODAUNACELLULALabel
```

```
app.PERCORSOCOMPIUTODAUNACELLULALabel = uilabel(app.GridLayout18);  
app.PERCORSOCOMPIUTODAUNACELLULALabel.HorizontalAlignment = 'center';  
app.PERCORSOCOMPIUTODAUNACELLULALabel.FontWeight = 'bold';  
app.PERCORSOCOMPIUTODAUNACELLULALabel.Layout.Row = 1;
```

```
app.PERCORSOCOMPIUTODAUNACELLULALabel.Layout.Column = 2;  
app.PERCORSOCOMPIUTODAUNACELLULALabel.Text = 'PERCORSO COMPIUTO DA UNA CELLULA';
```

```
% Create GridLayout19
```

```
app.GridLayout19 = uigridlayout(app.GridLayout18);  
app.GridLayout19.Layout.Row = 2;  
app.GridLayout19.Layout.Column = 1;
```

```
% Create LetteredellacolonnaexcelEditField
```

```
app.LetteredellacolonnaexcelEditField = uieditfield(app.GridLayout19, 'text');  
app.LetteredellacolonnaexcelEditField.ValueChangedFcn = createCallbackFcn(app, @lettera, true);  
app.LetteredellacolonnaexcelEditField.Layout.Row = 1;  
app.LetteredellacolonnaexcelEditField.Layout.Column = 2;
```

```
% Create ColonnaExcelLabel
```

```
app.ColonnaExcelLabel = uilabel(app.GridLayout19);  
app.ColonnaExcelLabel.Layout.Row = 1;  
app.ColonnaExcelLabel.Layout.Column = 1;  
app.ColonnaExcelLabel.Text = {'Colonna Excel'; ''};
```

```
% Create PercorsodellacellulaButton
```

```
app.PercorsodellacellulaButton = uibutton(app.GridLayout18, 'push');  
app.PercorsodellacellulaButton.ButtonPushedFcn = createCallbackFcn(app, @percorso, true);  
app.PercorsodellacellulaButton.Layout.Row = 2;  
app.PercorsodellacellulaButton.Layout.Column = 2;  
app.PercorsodellacellulaButton.Text = 'Percorso della cellula';
```

```
% Create ParametriavanzatiTab
```

```
app.ParametriavanzatiTab = uitab(app.TabGroup);  
app.ParametriavanzatiTab.Title = 'Parametri avanzati';  
app.ParametriavanzatiTab.ForegroundColor = [0 0 1];
```

```
% Create GridLayout20
```



```

app.GridLayout20 = uigridlayout(app.ParametriavanzatiTab);
app.GridLayout20.ColumnWidth = {'1x', 36, 82, '4x', '5.28x', 59, 52, '4x', '23.56x'};
app.GridLayout20.RowHeight = {'1x', 22, '1.08x', 22, '1.29x', 22, '2.38x'};

% Create NumerofotogrammiEditFieldLabel
app.NumerofotogrammiEditFieldLabel = uilabel(app.GridLayout20);
app.NumerofotogrammiEditFieldLabel.HorizontalAlignment = 'right';
app.NumerofotogrammiEditFieldLabel.Layout.Row = 2;
app.NumerofotogrammiEditFieldLabel.Layout.Column = [2 3];
app.NumerofotogrammiEditFieldLabel.Text = 'Numero fotogrammi';

% Create NumerofotogrammiEditField
app.NumerofotogrammiEditField = uieditfield(app.GridLayout20, 'numeric');
app.NumerofotogrammiEditField.ValueChangedFcn = createCallbackFcn(app, @Numerofotogrammi, true);
app.NumerofotogrammiEditField.Layout.Row = 2;
app.NumerofotogrammiEditField.Layout.Column = 4;

% Create RicercamitosiEditFieldLabel
app.RicercamitosiEditFieldLabel = uilabel(app.GridLayout20);
app.RicercamitosiEditFieldLabel.HorizontalAlignment = 'right';
app.RicercamitosiEditFieldLabel.Layout.Row = 4;
app.RicercamitosiEditFieldLabel.Layout.Column = 3;
app.RicercamitosiEditFieldLabel.Text = 'Ricerca mitosi';

% Create RicercamitosiEditField
app.RicercamitosiEditField = uieditfield(app.GridLayout20, 'numeric');
app.RicercamitosiEditField.ValueChangedFcn = createCallbackFcn(app, @Ricercamitosi, true);
app.RicercamitosiEditField.Layout.Row = 4;
app.RicercamitosiEditField.Layout.Column = 4;
app.RicercamitosiEditField.Value = 5;

% Create RicercacellulapadreEditFieldLabel
app.RicercacellulapadreEditFieldLabel = uilabel(app.GridLayout20);

```

```
app.RicercaCellulapadreEditFieldLabel.HorizontalAlignment = 'right';
app.RicercaCellulapadreEditFieldLabel.Layout.Row = 6;
app.RicercaCellulapadreEditFieldLabel.Layout.Column = [2 3];
app.RicercaCellulapadreEditFieldLabel.Text = 'Ricerca cellula padre';
```

```
% Create RicercaCellulapadreEditField
```

```
app.RicercaCellulapadreEditField = uieditfield(app.GridLayout20, 'numeric');
app.RicercaCellulapadreEditField.ValueChangedFcn = createCallbackFcn(app, @RicercaCellulapadre, true);
app.RicercaCellulapadreEditField.Layout.Row = 6;
app.RicercaCellulapadreEditField.Layout.Column = 4;
app.RicercaCellulapadreEditField.Value = 2;
```

```
% Create RicercaCellulefiglieEditFieldLabel
```

```
app.RicercaCellulefiglieEditFieldLabel = uilabel(app.GridLayout20);
app.RicercaCellulefiglieEditFieldLabel.HorizontalAlignment = 'right';
app.RicercaCellulefiglieEditFieldLabel.Layout.Row = 2;
app.RicercaCellulefiglieEditFieldLabel.Layout.Column = [6 7];
app.RicercaCellulefiglieEditFieldLabel.Text = 'Ricerca cellule figlie';
```

```
% Create RicercaCellulefiglieEditField
```

```
app.RicercaCellulefiglieEditField = uieditfield(app.GridLayout20, 'numeric');
app.RicercaCellulefiglieEditField.ValueChangedFcn = createCallbackFcn(app, @RicercaCellulefiglie, true);
app.RicercaCellulefiglieEditField.Layout.Row = 2;
app.RicercaCellulefiglieEditField.Layout.Column = 8;
app.RicercaCellulefiglieEditField.Value = 2;
```

```
% Create DistanzaEditFieldLabel
```

```
app.DistanzaEditFieldLabel = uilabel(app.GridLayout20);
app.DistanzaEditFieldLabel.HorizontalAlignment = 'right';
app.DistanzaEditFieldLabel.Layout.Row = 4;
app.DistanzaEditFieldLabel.Layout.Column = 7;
app.DistanzaEditFieldLabel.Text = 'Distanza';
```

```
% Create DistanzaEditField
```

```
app.DistanzaEditField = uieditfield(app.GridLayout20, 'numeric');  
app.DistanzaEditField.ValueChangedFcn = createCallbackFcn(app, @RicercaDistanza, true);  
app.DistanzaEditField.Layout.Row = 4;  
app.DistanzaEditField.Layout.Column = 8;  
app.DistanzaEditField.Value = 50;
```

```
% Create TempoEditFieldLabel
```

```
app.TempoEditFieldLabel = uilabel(app.GridLayout20);  
app.TempoEditFieldLabel.HorizontalAlignment = 'right';  
app.TempoEditFieldLabel.Layout.Row = 6;  
app.TempoEditFieldLabel.Layout.Column = 7;  
app.TempoEditFieldLabel.Text = 'Tempo';
```

```
% Create TempoEditField
```

```
app.TempoEditField = uieditfield(app.GridLayout20, 'numeric');  
app.TempoEditField.ValueChangedFcn = createCallbackFcn(app, @Tempofoto, true);  
app.TempoEditField.Layout.Row = 6;  
app.TempoEditField.Layout.Column = 8;  
app.TempoEditField.Value = 2700;
```

```
% Create IstruzioniTab
```

```
app.IstruzioniTab = uitab(app.TabGroup);  
app.IstruzioniTab.Title = 'Istruzioni';  
app.IstruzioniTab.BackgroundColor = [1 1 1];  
app.IstruzioniTab.ForegroundColor = [1 0 0];
```

```
% Create GridLayout13
```

```
app.GridLayout13 = uigriddisplay(app.IstruzioniTab);  
app.GridLayout13.ColumnWidth = {'1x', '11x', '1x'};  
app.GridLayout13.RowHeight = {'1x', '1x', '11x', '1x', '1x'};
```

```
% Create PremereperaprireilmanualedeleIstruzioniDellAlgoritmoButton
```

```

app.Premereperaprireilmanualedele istruzionidellalgoritmoButton = uibutton(app.GridLayout13, 'push');
app.Premereperaprireilmanualedele istruzionidellalgoritmoButton.ButtonPushedFcn = createCallback-
Fcn(app, @Istruzioni, true);
app.Premereperaprireilmanualedele istruzionidellalgoritmoButton.FontSize = 20;
app.Premereperaprireilmanualedele istruzionidellalgoritmoButton.FontWeight = 'bold';
app.Premereperaprireilmanualedele istruzionidellalgoritmoButton.FontColor = [1 0 0];
app.Premereperaprireilmanualedele istruzionidellalgoritmoButton.Layout.Row = 3;
app.Premereperaprireilmanualedele istruzionidellalgoritmoButton.Layout.Column = 2;
app.Premereperaprireilmanualedele istruzionidellalgoritmoButton.Text = 'Premere per aprire il manuale
delle istruzioni dell"algoritmo';

% Show the figure after all components are created
app.UIFigure.Visible = 'on';
end
end

% App creation and deletion
methods (Access = public)

% Construct app
function app = quasiquasi

% Create UIFigure and components
createComponents(app)

% Register the app with App Designer
registerApp(app, app.UIFigure)

% Execute the startup function
runStartupFcn(app, @start)

if nargin == 0

```

```
clear app
```

```
end
```

```
end
```

```
% Code that executes before app deletion
```

```
function delete(app)
```

```
% Delete UIFigure when app is deleted
```

```
delete(app.UIFigure)
```

```
end
```

```
end
```

```
end
```

CAPITOLO 5 - CONCLUSIONE

Ad oggi lo studio e l'analisi delle cellule tumorali sono molto complesse e costose a causa dei pochi strumenti finora scoperti. Lo sviluppo dell'algoritmo in questione può rendere molto più efficace tale analisi, permettendo una visione completa riguardo i loro movimenti e il loro tasso di divisione. Successivi esperimenti applicati a cellule tumorali in vitro (ad esempio l'utilizzo di campi elettromagnetici o di onde acustiche) necessitano di un metodo veloce e semplice che possa permettere uno studio approfondito della loro efficacia su tali cellule. La semplice e intuitiva interfaccia grafica permette l'utilizzo del programma a qualsiasi tipo di ricercatore, così da dare possibilità a molte più persone di entrare a far parte di questi studi con lo scopo di poterne comprendere la natura e di poter sviluppare terapie per pazienti sempre più efficienti e risolutive.

BIBLIOGRAFIA

1. Karp, G. *BIOLOGIA CELLULARE E MOLECOLARE*. (1998).
2. Hardin, J., Bertoni, G. P. & Kleinsmith, L. J. *BECKER - IL MONDO DELLA CELLULA*. (2014).
3. Fantoni, A., Bozzaro, S. & Del Sal, G. *BIOLOGIA CELLULARE E GENETICA*. (2009).
4. Lopez, M. *ONCOLOGIA MEDICA PRATICA*. (2005).
5. Scholzen, T. & Gerdes, J. The Ki-67 protein: From the known and the unknown. *Journal of Cellular Physiology* (2000).
6. Gerdes, J. *et al.* Cell cycle analysis of a cell proliferation-associated human nuclear antigen defined by the monoclonal antibody Ki-67. *J. Immunol.* (1984).