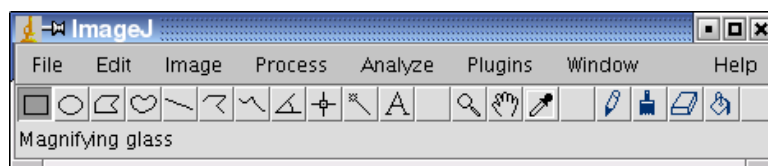


RotateProject und RotateDepthCode

Beschreibung

RotateProject und RotateDepthCode drehen einen Bilderstapel um eine vorgegebene Achse und erstellen bei jedem Rotationsschritt eine Projektion durch den Stapel. RotateDepthCode codiert die Tiefe der angezeigten Strukturen zusätzlich durch unterschiedliche Farben. Von jeder Projektion wird eine Bilddatei gespeichert. Aus den Bilddateien kann mit ImageJ oder Gimp eine Animation erzeugt werden (siehe [Programmbeschreibungen](#)), die die hilft, 3D Struktur zu klären.



Installation

RotateProject und RotateDepthCode können in den Macros-Ordner von ImageJ kopiert werden. Beide Makros benötigen die Stack-Plugins aus der [WCIF Plugin-Sammlung](#) und die [TransformJ Plugins](#).

Benutzung

Zur Benutzung müssen bei den Makros zwei Einstellungen vorgenommen werden:

1. Die Richtung und die Schrittweite der Drehung muss im Makro angegeben werden.
2. Die Pfade für das Verzeichnis mit den Quelldateien und den Zieldateien muss angegeben werden.

Richtung und die Schrittweite der Drehung

Zum Editieren wird mit `> Plugins > Edit` die Datei ausgewählt. In der 3. Zeile werden die Rotationsschritte angegeben:

```
for(i=5; i<=360; i=i+5){
```

`i=5` gibt an, dass das erste Bild bei 5 Grad Rotation erstellt werden soll, `i <= 360` gibt an, bis zu welchem Winkel rotiert werden soll und `i=i+5` gibt die Schrittweite in Grad an, hier 5 Grad.

In der Zeile:

```
run("TJ Rotate", "z-angle=0.0 y-angle="+i+" x-angle=0.0 scheme=4  
filling=0");
```

wird angegeben, um welche Achse der Bildstapel rotiert werden soll. Hinter den entsprechenden Achsen-Winkel `z-angle=` wird `" +i+ "` eingetragen, so dass jeweils um den oben angegebenen `i`-Wert rotiert wird. Achsen, um die nicht rotiert werden soll, werden auf `0.0` gesetzt.

Pfade für Quell- und Zielverzeichnis

Quellverzeichnis: Beide Makros fragen nach dem Start, welchen Bilderstapel sie öffnen sollen. Der Name des Verzeichnisses in dem die Bilder liegen, wird der Name des ersten Fensters. Da dieses Fenster vom Makro immer wieder angesprochen wird, ist es wichtig, dass der Name dem Makro bekannt ist. Im Skript ist der Name des Verzeichnisses mit `work` angegeben: z.B.: `selectWindow("work");` oder `while (isOpen("work rotated") == false){`. Es bietet sich an, den Bilderstapel der bearbeitet werden soll, in ein Verzeichnis `work` zu legen, wobei egal ist, wo sich dieses Verzeichnis befindet. Als Alternative kann im Skript jedes Vorkommen

von `work` gegen den gewünschten Verzeichnisnamen ausgetauscht werden, ich warne hiermit aber ausdrücklich vor Verzeichnisnamen mit Leerzeichen, Punkten, Kommata und anderen Spielereien. **Wichtig:** alte Versionen von ImageJ bezeichnen den entstandenen Stapel statt mit `work rotated` mit `Rotated work`. Dies sieht man an der Bezeichnung des Fensters, das nach der ersten Rotation entsteht. Wenn das der Fall ist, muss im Skript überall `Rotated` **vor** statt hinter den Verzeichnisnamen gesetzt und groß geschrieben werden.

Zielverzeichnis: Das Zielverzeichnis wird in der Zeile:

```
saveAs("Tiff", "C:\\Dokumente und Einstellungen\\bquast\\Eigene  
Dateien\\projections\\Proj" + i + ".tif");
```

angegeben. „Tiff“ beschreibt den Dateityp, `"C:\\Dokumente und Einstellungen\\bquast\\Eigene Dateien\\projections\\Proj" + i + ".tif"` den Pfad und die Dateinamen, unter denen die Dateien gespeichert werden. Der doppelte Backslash `\\` sorgt dafür, dass der zweite der beiden Backslashes als solcher erkannt wird und nicht als Anweisung im Programm interpretiert wird.

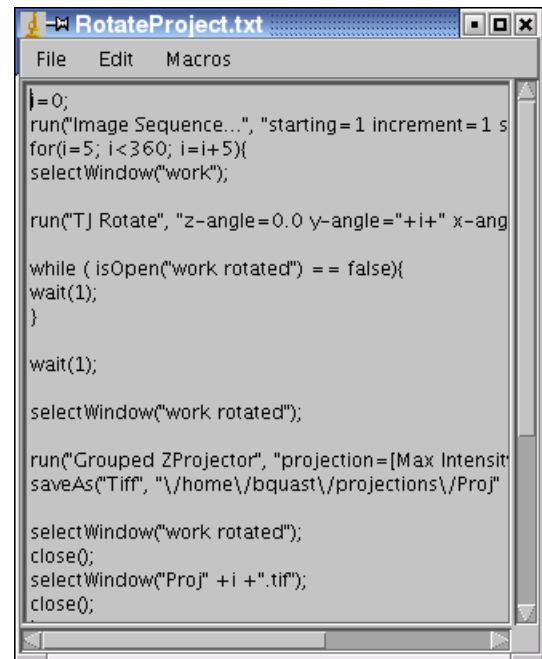
Unter Linux würde der Pfad etwa so eingegeben:

```
"\\home\\bquast\\projections\\Proj" + i + ".tif"
```

auch hier müssen das ein Sonderzeichen wie der Slash `/` durch einen vorherigen Backslash kenntlich gemacht werden.

Nachdem der Pfad gesetzt ist, können die Makros mit `> File > Save` abgespeichert und mit `> File > Run Macro` gestartet werden. In der neuesten Version von ImageJ befindet sich `> Run Macro` unter dem Menüpunkt `> Macros`.

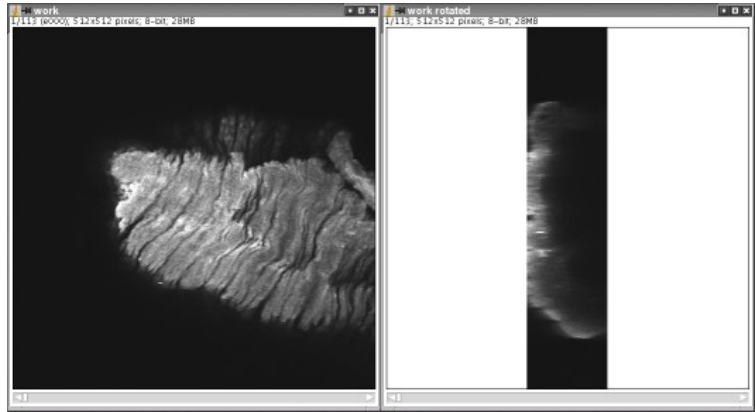
Nach dem Starten öffnet sich ein Dialog zum Öffnen des Bildstapels. Es kann und muss nur eine Datei angegeben werden. ImageJ nimmt alle Dateien aus dem Verzeichnis, daher ist es sinnvoll, nur Dateien die zum Stapel gehören in diesem Verzeichnis zu haben.



Tipps

Je nach Größe der Dateien kann die Berechnung einige Stunden dauern und größere Mengen Arbeitsspeicher in Anspruch nehmen. Der für ImageJ zur Verfügung stehende Arbeitsspeicher kann im ImageJ Hauptfenster unter **Edit > Options > Memory** eingestellt werden.

Bevor man die Berechnung startet, ist es sinnvoll vor und hinter dem Stapel so viele Ebenen in der Hintergrundfarbe einzufügen, dass der Stapel auch beim Drehen um 90 Grad das gesamte Bild ausfüllt. Ein Stapel mit 100 Pixeln Breite (X-Achse), sollte wenn er um die Y-Achse gedreht wird auch 100 Ebenen haben, da sonst links und rechts weiße Ränder entstehen.



Ebenen können mit den Stacks-Building und Stacks-Shuffling Plugins eingefügt und verschoben werden.