

RIJKSINSTITUUT VOOR VISSERIJONDERZOEK

Haringkade 1 - Postbus 68 - 1970 AB IJmuiden - Tel.: +31 2550 64646

Afdeling: Pelagische Zeevisserij

Rapport: PELVIS 90-02

Toepassingsmogelijkheden image-analysis
bij meting haringlarven en fecunditeitsbepaling
horsmakreel; een oriënterend onderzoek".

Auteur(s): J. Glorie

Project: 40.013 en 40.021

Projectleider: A. Corten

Datum van verschijnen: juni 1990

Inhoud:

- 1 Lengtemeting haringlarven met behulp van TCL-image
- 2 Computer programma "Herringlarval"
- 3 Fecunditeitsbepaling horsmakreel met behulp van TCL-image
- 4 Computerprogramma "Horsmakreel"

Lengte-meting haringlarven met behulp van TCL-Image.

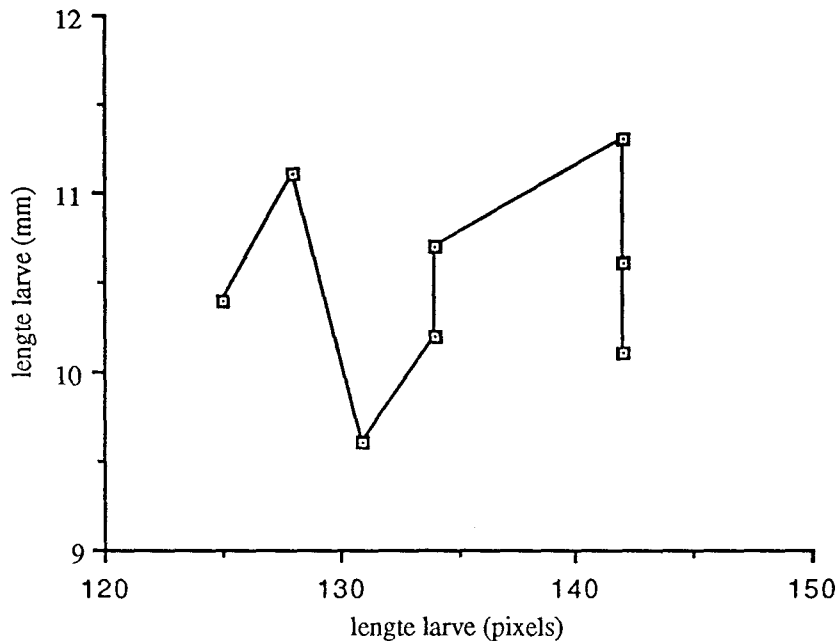
Voor de lengte meting van haringlarven met behulp van TCL-Image is een basisprogramma geschreven. Dit programma leest een opname met haringlarven in. Op deze opname wordt een Kuwahara filter toegepast. Daarna wordt op het ontstane beeld twee verschillende thresholds gezet. Deze twee thresholds leveren twee verschillende bitplanes(iedere pixel kan de waarde 0 of 1 aannemen), door deze bitplanes nu met elkaar te laten propageren wordt een nieuw bitplane gevormd met de gevonden objecten(haringlarven en zeer intensieve achtergrond). Na deze propagatie volgt nog een selectie op grootte. Hierdoor worden de objecten die geen haringlarven zijn verwijderd. De laatste stap is het skellettoneren van de larven. Als er een skelet van de larven is gemaakt kan er een lengte meting gedaan worden. Dit gedeelte van het programma moet nog verder uitgewerkt worden.

Er zijn enkele metingen verricht waarbij eerst de lengte van de larve is gemeten met behulp van een maatschaal onder de microscoop en daarna met het haringlarveprogramma. De resultaten volgen in tabel 1.

Tabel 1.

lengte larve (mm)	lengte larve (pixels)
11,3	142
9,6	131
10,2	134
11,1	128
10,6	142
10,7	134
10,4	125
10,1	142

Uitgezet in een grafiek geven deze metingen het volgende beeld.



Problemen

1. Het kuwahara filter heeft zoveel geheugen ruimte nodig dat er niet met opnames van $512 * 512$ pixels gewerkt kan worden. Dit brengt met zich mee dat er maar enkele haringlarven(2 of 3) in een opname van $256 * 256$ pixels aanwezig zijn.
2. De haring larven moeten netjes naast elkaar in de opname liggen. Ze mogen niet tegen of over elkaar heen liggen.
3. Het skelet mist soms nog enkele pixels aan de kop en/of aan de staart. Dit geeft een afwijking.

Deze problemen kunnen misschien opgelost worden door verdere ontwikkeling van het programma. Er moet dan een betere manier gevonden worden om een zo goed mogelijk skelet te verkrijgen. Het kuwahara filter moet verwijderd worden omdat dit teveel geheugen ruimte inneemt. Dan is het mogelijk om met $512 * 512$ opnamen te gaan werken(meerdere larven in een opname).

Om ongeveer 20 larven in een opname te krijgen moet er met een kleinere vergroting gewerkt worden. Dit betekent dat er een nieuwe microscoop aangeschaft moet worden wat weer extra kosten met zich meebrengt.

Conclusie:

Het is waarschijnlijk mogelijk om een programma te maken dat haringlarven kan meten. Om dit programma 100 % te krijgen moet er nog erg veel tijd(meerdere maanden) besteed worden aan het verbeteren en afmaken van het nu bestaande basisprogramma.

De vraag of de computer het sneller kan dan een analist moet op dit moment met nee beantwoord worden, omdat het maken van een opname nogal wat tijd vergt.

De vraag of de computer het nauwkeuriger kan dan een analist kan momenteel nog niet beantwoord worden, omdat het basisprogramma nog niet goed genoeg werkt.

*****HERRINGLARVAL*****

* *

** This program measured the length of
** herring larval

* *

* *

***** ® By Jan Glorie *****

* *

* *

* *

Kpool ! verwijderen van voor gedeclareerde variabelen

*****Declaration of the variables*****

decl/world orig short 256 256

decl/world r1 short 256 256

decl/world r2 short 256 256

decl/world bitpl short 256 256

decl/world/position=bitpl bit1 sbit/plane=1 256 256

decl/world/position=bitpl bit2 sbit/plane=2 256 256

decl/world/position=bitpl bit3 sbit/plane=3 256 256

decl/world/position=bitpl bit4 sbit/plane=4 256 256

decl/world/position=bitpl bit5 sbit/plane=5 256 256

decl/world/position=bitpl bit6 sbit/plane=6 256 256

*****Reading file*****

readf/binary ? orig

disp/v=1/h='original' orig

*****Kuwahara filter*****

kuwa orig r1 5

disp/v=2/h='kuwa' r1

wait

*****Thresholding file*****

thresh r1 bit1 f 170

binv bit1 bit1

disp/v=3/h='treshold170'/red=bit1 orig

```
thresh r1 bit2 f 110
binv bit2 bit2
disp/v=4/h='treshold 110'/red=bit2 orig
```

```
*****Propagation*****
```

```
bprop bit2 bit1 bit3
disp/v=5/h='propagation'/red=bit3 orig
```

```
*****Sos hull*****
```

```
blabel bit3 r2 8
sos r2 r2 500
rhull r2 r2 15
thresh r2 bit4 f 1
disp/v=6/h='sos hull'/red=bit4 orig
```

```
*****Skelletoneren*****
```

```
bremh bit4
bdist bit4 r2
bdskel r2 bit5
disp/v=6/h='skelletoned'/red=bit5 orig
```

```
*****Lenght measuring*****
```

```
declare labels int
```

```
blabel bit5 r2 8 labels
dens r2 orig
```

Fecunditeitsbepaling horsmakreel met behulp van TCL-Image.

Voor de fecunditeitsbepaling met behulp van TCL-Image van horsmakreel is een basis programma geschreven. Dit programma leest een opname van 256*256 pixels in. In deze opname moet het programma de eicellen die in ontwikkeling zijn en de eicellen die zich niet ontwikkelen gescheiden tellen. Om dit te bereiken worden op de opname drie verschillende thresholds gezet. Hierdoor worden drie verschillende bitplanes gevormd, een bitplane als uitgangsbithplane, een bitplane voor het verwijderen van achtergrondruis en een bitplane dat wordt gebruikt als masker voor een propagatie. Door in het uitgangsbithplane te selecteren op minimale (2) en maximale (500) grootte worden de eicellen gevonden (ontwikkeld en niet ontwikkeld). Nu volgt de scheiding en telling van de eicellen. De niet ontwikkelde eicellen zijn voor ongeveer 90 % te vinden (grijswaarde verschillen zijn te klein). Ontwikkelde eicellen zijn te vinden als meerdere kleine objecten in een bitplane dit levert moeilijkheden op bij het tellen van de eicellen omdat een eicel meerdere keren geteld wordt. Er is geprobeerd om dit op te vangen door middel van een propagatie met een ander bitplane en door middel van een dilation (uitzetten van de objecten) maar dat bracht ook niet het gewenste resultaat.

Conclusie

Het maken van een programma in TCL-Image voor de fecunditeitsbepaling is zeer moeilijk (wellicht onmogelijk). Dit komt omdat de grijswaarden in de opname allemaal zeer dicht bij elkaar liggen. Een scheiding van de objecten wordt hierdoor zeer moeilijk. Dit gaat betekenen dat er al heel snel objecten worden geteld die geen eicellen zijn of er objecten worden gemist die wel meegeteld hadden moeten worden. Er kan ook niet op vorm worden geselecteerd omdat, iedere eicel weer een andere vorm bezit (groot, klein, enz.). Om te werken met 512*512 opnamen zal ook hier een geheugen uitbreiding nodig zijn. Er kunnen dan wel vier keer zo veel cellen bekeken worden als nu het geval was.

```

*****Horsmakreel*****
* *
* *
*****Door Jan Glorie*****
* *
* *
* *

```

Kpool ! verwijderen van voor gedeclareerde variabelen

```
*-----Declarations
```

```

decl/world orig short 256 256
decl/world r1 short 256 256
decl/world r2 short 256 256
decl/world r3 short 256 256

```

```

decl/world bitpl short 256 256
decl/world/position=bitpl bit1 sbit/plane=1 256 256
decl/world/position=bitpl bit2 sbit/plane=2 256 256
decl/world/position=bitpl bit3 sbit/plane=3 256 256
decl/world/position=bitpl bit4 sbit/plane=4 256 256
decl/world/position=bitpl bit5 sbit/plane=5 256 256
decl/world/position=bitpl bit6 sbit/plane=6 256 256
decl/world/position=bitpl bit7 sbit/plane=7 256 256

```

```

!declare teller1 int
!declare teller2 int
!declare ontwikkeld int
!declare nietontwikkeld int

```

```

!copy 0 teller1
!copy 0 teller2

```

```

!write/outfile=helpfile1 teller1
!write/outfile=helpfile2 teller2

```

```
!for tclhm2 to tclhm2 do
```

```
log
```

```
*-----Reading image
```

```

readf/binary tclhm2 orig
disp/v=1/h='original' orig

```



```
*-----Thresholding image
!0 zwart, 256 wit
thresh orig bit1 f 185 !bitplane waarmee gewerkt wordt.
thresh orig bit2 f 195 !bitplane voor verwijderen ruis.
thresh orig bit3 f 180 !bitplane als masker voor propagatie.
```

```
disp/v=2/h='thresh 185'/red=bit1 orig
```

```
*-----niet ontwikkelde eieren
```

```
label bit1 r1 8
```

```
sos r1 r2 2
```

```
thresh r2 bit4 f 1
```

```
!disp/v=3/h='Midden en grote objecten'/red=bit4 orig
```

```
sos r1 r3 500
```

```
thresh r3 bit5 f 1
```

```
disp/v=4/h='Grote objecten'/red=bit5 orig
```

```
bxor bit4 bit5 bit5
```

```
disp/v=5/h='Midden objecten'/red=bit5 orig
```

```
label bit5 r1 8 !nietontwikkeld
```

```
*-----Ontwikkelde eieren
```

```
bxor bit1 bit4 bit4 !alleen kleine objecten
```

```
disp/v=6/h='kleine objecten'/red=bit4 orig
```

```
band bit2 bit4 bit6 !verwijderen ruis
```

```
disp/v=7/h='minus achtergrond'/red=bit6 orig
```

```
*-----Propagation
```

```
disp/v=8/h='thresh 180'/red=bit3 orig
```

```
bprop bit6 bit3 bit6 !ontwikkelde eieren
```

```
disp/v=9/h='propagation'/red=bit6 orig
```

```
!na propagatie wordt weer geselecteerd op groote, want kleine en
```

```
!grote objecten zijn geen ontwikkelde eieren
```

```
label bit6 r1 8
```

```
sos r1 r2 2          !andere keuzen
thresh r2 bit6 f 1
sos r1 r3 1000      !andere keuzen
thresh r3 bit7 f 1
bxor bit6 bit7 bit6
disp/v=2/h='ontwikkeld'/red=bit6 orig
```

```
!qpix bit6
```

```
blabel bit6 r1 8 !ontwikkeld
```

```
!read/infile=helpfile1 teller1
!read/infile=helpfile2 teller2
!add nietontwikkeld teller1
!add ontwikkeld teller2
!write/outfile=helpfile1 teller1
!write/outfile=helpfile2 teller2
```

```
!end [for]
```

```
!read/infile=helpfile1 teller1
!read/infile=helpfile2 teller2
!write teller1
!write teller2
```