Parte V

Apéndices

Apéndice A

Algoritmos empleados

A.1. image2radi.m

```
1  %image2radi.m                                  Carlos Lopez Roa CFATA-UNAM 2013
2  % Este codigo toma carpetas de fotografias en formato .tif y las convierte
3  % en arreglos de valores numericos en .csv
4  %% init
5  tic    %Inicia un contador de tiempo
6  clc    %Limpiar prompt
7  clear all %Limpiar Workspace
8  cd /Users/Poincare/Dropbox/Documentos/CFaTa/8CLR/ServicioSocial/
9  %Cambiar de directorio activo
10 v=131000;                                        %Velocidad de las grabaciones
11 pre=['B0';'F0';'D0';'E0';'EA';'EC';'ED';'EE';'EF';'EG';'EH';'EI';'EJ';'EK';
12      'EL';'EM';'EN';'EO';'EP';'EQ';'ER'];       %Subcarpetas a explorar
13 for im=1:length(pre) %Ciclo general sobre las subcarpetas
14     mat='/Users/Poincare/Desktop/Datos4/'; %El directorio donde se alojan
15     %las subcarpetas
16     prefix=pre(im,:);                           %Cambiador de subcarpetas
17     switch prefix
18         case 'B0'
19             prefix='B';
20         case 'E0'
21             prefix='E';
22         case 'D0'
23             prefix='D';
24         case 'F0'
25             prefix='F';
26     end
27     cd (strcat('/Users/Poincare/Dropbox/MainWavelet/',prefix))
28     %Cambia de directorio a donde se quiere escribir los archivos CSV%
```

```
29      dd=360;       %270,380,590  %368          %Fotograma de inicio de interes
30      uu=dd+300;    %70,90,125  %%220           %Fotograma de fin de interess
31      jj=10;                                    %Iterador de los ensayos
32      maxi=59;                                  %Numero total de ensayos
33      ext='.tif';                               %Extension de las fotografias
34      time=[0:(1/v):(uu-dd-1)*(1/v)];           %Arreglo con los instantes
35      %correspondientes de tiempo real a cada fotograma
36      hold off                                  %Manejo de figuras
37      %% Plot data
38      switch prefix
39          case 'B'
40              mat='/Users/Poincare/Desktop/Datos2/';
41              dd=266+5;
42              uu=dd+215;
43              maxi=15;
44              jj=1;
45              prefix1='ImgB';
46          case 'E'
47              mat='/Users/Poincare/Desktop/Datos2/';
48              dd=380;
49              uu=dd+300;
50              maxi=99;
51          case 'D'
52              mat='/Users/Poincare/Desktop/Datos2/';
53              dd=266+5;
54              uu=dd+215;
55              maxi=99;
56          case 'F'
57              mat='/Users/Poincare/Desktop/Datos2/';
58              dd=605-22;
59              uu=dd+415;
60              maxi=99;
61
62      end
63
64      while jj<=maxi                            %Segundo ciclo sobre los ensayos
65          dire=strcat(mat,prefix,'0',int2str(jj),'/'); %Directorio del ensayo
66          if jj<10
67              dire=strcat(mat,prefix,'00',int2str(jj),'/');
68          end
69          j=dd;                                     %Iterador de los fotogramas
70          r=zeros(uu-dd,1);
71          while j<uu                          %Tercer ciclo sobre los fotogramas
72              name=strcat(dire,prefix,'000',int2str(j),ext);
73              if prefix=='B'
74                  name=strcat(dire,prefix1,'000',int2str(j),ext);
75              end
76
77              %Nombre del fotograma
78              cdata=imread(name);          %Variable de los fotogramas
```

```
79              dim=length(cdata);              %Dimension del arreglo del fotograma
80              cdata=255-cdata;                %Inversion
81              cdata=cast(cdata,'double');%Hace cdata un arreglo tipo doble
82              j=j+1;                          %Aumento de interador del fotograma
83              r(j-dd)=sqrt(mean(mean(cdata))); %Aplicacion de la funcion f
84          end
85          %r=r/r(1);                          %Se adimensionaliza r
86          plot(r)                             %Grafica r
87          grid on                             %Activa la malla
88          hold on                             %Manejo de figuras
89          clc
90          prefix
91          jj
92
93          %pause                              %Pausa optativa
94          %Export to Mathematica
95          xlswrite(strcat('r',prefix,int2str(jj)),r) %Exporta r en CSV
96          jj=jj+1;                            %Aumento de interador de ensayos
97      end
98      hold off
99      %pause
100 end
101 hold off                                    %Manejo de figuras
102 cd /Users/Poincare/Dropbox/Documentos/CFATA/8CLR/ServicioSocial/
103 toc                                  %Muestra el tiempo empleado en el computo
```

A.2. mainwavelet.nb

```
1  (*Constantes*)
2  pre = {"B", "F", "D", "E", "EA", "EC", "ED", "EE", "EF", "EG", "EH",
3     "EI", "EJ", "EK", "EL", "EM", "EN", "EO", "EP", "EQ", "ER"};
4  root = "/Users/Carlos/Desktop/Dropbox"; folder1 = "MainWavelet"; \
5  name1 = "r";
6  If[$MachineName == "Poincare" || $MachineName == "poincare",
7    root = "/Users/Poincare/Dropbox/",
8    root = "/Users/Carlos/Desktop/Dropbox"];
9  (*Funciones*)
10 NoisyExtremaFinder = Function[{valueList, aroundRange},
11    timeList = Range[Length[valueList]];
12    extremaPosition =
13     Flatten@Position[
14        Map[#, Partition[valueList, 2*aroundRange + 1,
15            1, {-(1 + aroundRange), 1 + aroundRange}, {}]] - valueList,
16         0.] &;
17    extremaPoints =
18     Transpose@{timeList[[#]], valueList[[#]]} &@extremaPosition[#] &;
19    {extremaPoints[Min], extremaPoints[Max]}];
20 NoisyThreshPeaks[ei_, vecinity_, tresh_] := (
```

```
21      ntp = DeleteCases[
22        Table[
23         Select[
24          NoisyExtremaFinder[ei, vecinity][[i]],
25          Abs[#[[2]]] > Abs[tresh*Max[ei]] ||
26            Abs[#[[2]]] > Abs[tresh*Min[ei]] &]
27         , {i, 1, Length[NoisyExtremaFinder[ei, vecinity]]}], {}];
28      Return[ntp]
29      );
30  FreqId[oo_] := (
31    pper = N[
32      Length[oo]/(Position[Abs[Fourier[oo]], Max[Abs[Fourier[oo]]][[1,
33           1]] - 2 +
34          2 (Position[
35                Abs[Fourier[
36                  oo Exp[2 Pi I (Position[Abs[Fourier[oo]],
37                       Max[Abs[Fourier[oo]]][[1, 1]] - 2) N[
38                       Range[0, Length[oo] - 1]]/Length[oo]],
39                  FourierParameters -> {0, 2/Length[oo]}]],
40                Max[Abs[
41                  Fourier[
42                   oo Exp[2 Pi I (Position[Abs[Fourier[oo]],
43                       Max[Abs[Fourier[oo]]][[1, 1]] - 2) N[
44                       Range[0, Length[oo] - 1]]/Length[oo]],
45                   FourierParameters -> {0, 2/Length[oo]}]][[1, 1]] -
46              1)/Length[oo])];
47    Return[pper]
48    )
49  obs[prefi_, nu_] := (
50      SetDirectory[FileNameJoin[{root, folder1}]];
51      prefix = prefi;
52      num = nu;
53      SetDirectory[FileNameJoin[{root, folder1, prefix}]];
54      rm = Flatten[
55        Import[StringJoin[name1, prefix, ToString[num], ".csv"]]];
56      SetDirectory[FileNameJoin[{root, folder1}]];
57      Return[rm]
58      );
59  filtrar[obsi_] := (
60      rf = InverseWaveletTransform[
61        WaveletThreshold[
62         StationaryWaveletTransform[obsi, SymletWavelet[], 3],
63         "VisuShrink"]];
64      Return[rf]
65      );
66  interpolar[obsi_] := (
67      ri = interpolar[obsi, .1];
68      Return[ri]
69      );
70  interpolar[obsi_, k_] := (
```

```
    ri = Table[ListInterpolation[obsi][[i]], {i, 1, Length[obsi], k}];
    Return[ri]
    );
normalizar[obsi_] := (
    rn = obsi/obsi[[1]];
    Return[rn]
    );
tocsv[obsi_] := (
    SetDirectory[FileNameJoin[{root, folder1, "fil"}]];
    obslocal = obsi;
    DumpSave[StringJoin["obs", ToString[prefix], ToString[num], ".mx"],
      obslocal];
    SetDirectory[FileNameJoin[{root, folder1, "fil"}]];
    );
fromcsv[prefi_, nu_] := (
    SetDirectory[FileNameJoin[{root, folder1, "fil"}]];
    Get[StringJoin["obs", ToString[prefi], ToString[nu], ".mx"]];
    Return[obslocal];
    );
treat[obsi_] := (
    trat = normalizar[filtrar[obsi]];
    Return[trat]
    );
paraest[obsi_] := (
    paraest[obsi, SymletWavelet[], 12]
    );
paraest[obsi_, wave_] := (
    paraest[obsi, wave, 12]
    );
paraest[obsi_, wave_, level_] := (
    swt = StationaryWaveletTransform[obsi, wave, level];
    deltashot =
     NoisyThreshPeaks[swt[All, "Values"][[2*6]], 10, .3][[2]][[All, 1]];
    edist = compar[obsi, interpolar[fromcsv["E", 10]]];
    f1 = FreqId[swt[All, "Values"][[2*8]]];
    peakf1 = NoisyThreshPeaks[swt[All, "Values"][[2*8]], 10, .3][[2]];
    f2 = FreqId[swt[All, "Values"][[2*9]]];
    peakf2 = NoisyThreshPeaks[swt[All, "Values"][[2*9]], 10, .3][[2]];
    pars = {deltashot, edist, f1, f2, peakf1, peakf2};
    Return[swt]
    );
compar[o1_, o2_] := (
    Return[
     Abs[Norm[ListConvolve[o1, o2, {1, 1}]] -
       1/2 (Norm[ListConvolve[o1, o1, {1, 1}]] +
          Norm[ListConvolve[o2, o2, {1, 1}]])]]
    );
```

A.3. defigure.nb

```
thesisdir =
  "/Users/Poincare/Dropbox/Documentos/CFATA/8CLR/ServicioSocial/\
Thesis";
local = normalizar[obs["F", 10]];
local1 = normalizar[obs["E", 10]];
local2 = normalizar[obs["D", 10]];
local = Table[{N[i/200], local[[i]]}, {i, 1, Length[local], 1}];
local1 = Table[{N[i/131], local1[[i]]}, {i, 1, Length[local1], 1}];
local2 = Table[{N[i/94], local2[[i]]}, {i, 1, Length[local2], 1}];
rr1 = ListLinePlot[{local, local1, local2},
  PlotLegends -> {"\!\(\*SubscriptBox[\(r\), \(200  K\)]\)",
    "\!\(\*SubscriptBox[\(r\), \(131  K\)]\)",
    "\!\(\*SubscriptBox[\(r\), \(94  K\)]\)"},
  AxesLabel -> {"t[ms]", "r[u.a.]"}, PlotStyle -> Thick]
SetDirectory[thesisdir]
Export["rr1.eps", rr1]
localarray = Table[treat[obs["B", i]], {i, 1, 15, 1}];
rr2 = ListPointPlot3D[localarray,
  AxesLabel -> {"t[u.a.]", "E[u.a.]", "r[u.a.]"}, Boxed -> False,
  ColorFunction -> "BlueGreenYellow"(*,Filling->Bottom*)]
rr2b = ListContourPlot[localarray, PlotLegends -> Automatic,
  ColorFunction -> "SunsetColors",
  FrameLabel -> {"t[u.a.]", "E[u.a.]"}]
SetDirectory[thesisdir]
Export["rr2.eps", rr2]
Export["rr2b.pdf", rr2b]
dial = {0, "200K", "94K", "131K", 200, 470, 335, 267, 402, 233, 301,
   368, 436, 216, 250, 284, 318, 351, 385, 419, 453};
pre2 = Table[{pre[[i]], dial[[i]]}, {i, 1, 4}];
pre1 = Table[{pre[[i]], dial[[i]]}, {i, 5, Length[pre]}];
ordnum = {1, 10, 6, 11, 4, 12, 7, 13, 3, 14, 8, 15, 5, 16, 9, 17, 2};
radigroup1 =
 GraphicsColumn[
  Table[ListLinePlot[Table[fromcsv[p, i], {i, 10, 99, 1}],
    PlotLabel -> pre2[[Flatten[Position[pre, p]]]][[1]]], {p,
    pre[[2 ;; 4]]}], ImageSize -> {300, Automatic}]
egroup = Table[
   ListLinePlot[Table[fromcsv[p, i], {i, 10, 59, 1}],
    PlotLabel -> pre1[[Flatten[Position[pre, p] - 4]]][[1]],
    PlotRange -> All], {p, pre[[5 ;; Length[pre]]]}];
radigroup2 = GraphicsGrid[{
   Table[egroup[[i]], {i, ordnum[[1 ;; 3]]}],
   Table[egroup[[i]], {i, ordnum[[4 ;; 6]]}],
   Table[egroup[[i]], {i, ordnum[[7 ;; 9]]}],
   Table[egroup[[i]], {i, ordnum[[10 ;; 12]]}],
   Table[egroup[[i]], {i, ordnum[[13 ;; 15]]}],
   Table[egroup[[i]], {i, ordnum[[16 ;; 17]]}]
   }, ImageSize -> {500, Automatic}]
```

```
SetDirectory[thesisdir]
Export["radigroup1.eps", radigroup1]
Export["radigroup2.eps", radigroup2]
eaeadif =
  DeleteCases[
   DeleteCases[
    ParallelTable[
     compar[fromcsv["EA", i], fromcsv["EA", j]], {i, 10, 59, 1}, {j,
      10, 59, 1}], 0., 2], Indeterminate, 2];
eedif = DeleteCases[
   DeleteCases[
    ParallelTable[
     compar[fromcsv["E", i], fromcsv["E", j]], {i, 10, 99, 1}, {j, 10,
       99, 1}], 0., 2], Indeterminate, 2];
ffdif = DeleteCases[
   DeleteCases[
    ParallelTable[
     compar[fromcsv["F", i], fromcsv["F", j]], {i, 10, 99, 1}, {j, 10,
       99, 1}], 0., 2], Indeterminate, 2];
dddif = DeleteCases[
   DeleteCases[
    ParallelTable[
     compar[fromcsv["F", i], fromcsv["F", j]], {i, 10, 99, 1}, {j, 10,
       99, 1}], 0., 2], Indeterminate, 2];
monodiftab =
  Table[DeleteDuplicates[
    Sort[Flatten[dif]]], {dif, {eedif, ffdif, dddif}}];
eediffx =
 GraphicsRow[
  Table[ArrayPlot[Log[#] &@{ffdif, dddif, eedif}[[i]],
    PlotLegends -> Automatic, ColorFunction -> "SolarColors",
    PlotLabel -> pre[[i + 1]]], {i, 1, Length[monodiftab]}],
  ImageSize -> {800, Automatic}]
eediff1 =
 Histogram[Log[10, #] &@monodiftab, ChartLayout -> "Stacked",
  ChartLegends -> {"E-E", "F-F", "D-D"},
  AxesLabel -> {"Log[d(G_i,G_j)]", "F"}]
in = AbsoluteTime[];
alldiff = DeleteCases[
   DeleteCases[
    Table[
     ParallelTable[
      compar[
       fromcsv[p, i], fromcsv[p, j]]
      , {i, 10, 59, 1}, {j, 10, 59, 1}], {p,
      pre[[5 ;; Length[pre]]]}], 0., 3], Indeterminate, 3];
out = AbsoluteTime[];
out - in
alldifftab =
  Table[DeleteDuplicates[Sort[Flatten[alldiff[[i]]]]], {i, 1,
```

```
    Length[alldiff]}];
eediff2 =
 Histogram[Log[10, #] &@alldifftab, ChartLayout -> "Stacked",
  ChartLegends -> pre[[5 ;; Length[pre]]],
  AxesLabel -> {"Log[d(G_i,G_j)]", "F"}, ImageSize -> {400, Automatic}]
arrayplotab =
  Table[ArrayPlot[Log[#] &@alldiff[[i]], PlotLegends -> Automatic,
    ColorFunction -> "SolarColors", PlotLabel -> pre[[i + 4]]], {i, 1,
     Length[alldiff]}];
eediff3 = GraphicsGrid[{
   arrayplotab[[1 ;; 4]],
   arrayplotab[[5 ;; 8]],
   arrayplotab[[9 ;; 12]],
   arrayplotab[[13 ;; 16]]}, ImageSize -> {800, Automatic}
  ]
SetDirectory[thesisdir]
Export["eediff1.pdf", eediff1]
Export["eediff2.pdf", eediff2]
Export["eediff3.pdf", eediff3]
Export["eediffx.pdf", eediffx]
in = AbsoluteTime[];
epars = ParallelTable[paraest[interpolar[fromcsv["E", i]]];
   pars, {i, 10, 99}];
expars = ParallelTable[paraest[interpolar[fromcsv[p, i]]];
   pars, {i, 10, 59}, {p, pre[[5 ;; Length[pre]]]}];
out = AbsoluteTime[];
out - in
SetDirectory[NotebookDirectory[]]
DumpSave["epars.mx", epars];
SetDirectory[NotebookDirectory[]]
<< "epars.mx"
parsestfig1 = DistributionChart[{
   Flatten[expars[[All, All, 1]]],
   Select[Flatten[epars[[All, 1]]], NumberQ[#] &]
   }, BarOrigin -> Left, ChartLabels -> {"EA-ER", "E"},
  ChartStyle -> "NeonColors"]
var1 = DeleteCases[Sort[Select[epars[[All, 2]], NumberQ]], 0.];
var2 = Sort[Flatten[expars[[All, All, 2]]]];
parsestfig2 = Histogram[{
   var2,
   var1
   }, ChartLegends -> {"EA-ER", "E"}, ChartStyle -> "NeonColors",
  ChartLayout -> "Stacked"]
var1 = Select[epars[[All, 3]], NumberQ];
var2 = Flatten[expars[[All, All, 3]]];
var3 = Select[epars[[All, 4]], NumberQ];
var4 = Select[Flatten[expars[[All, All, 4]]], # < 900 &];
parsestfig3 =
 DistributionChart[
  Reverse@{Sort[var1], Sort[var2], Sort[var3], Sort[var4]},
```

```
  BarOrigin -> Left, ChartStyle -> "NeonColors",
  ChartLabels ->
   Reverse@{"\!\(\*SubscriptBox[\(F\), \(8\)]\)E",
     "\!\(\*SubscriptBox[\(F\), \(8\)]\)EA–EX",
     "\!\(\*SubscriptBox[\(F\), \(9\)]\)E",
     "\!\(\*SubscriptBox[\(F\), \(10\)]\)EA–EX"}]
parsestfig4 = DistributionChart[
  Reverse@{
    FindClusters[
     Select[Flatten[Thread[Transpose[epars[[All, 5]]]][[All, 1]]],
      NumberQ[#] &], 6, Method -> "Agglomerate"],
    FindClusters[
     Select[Flatten[Thread[Transpose[epars[[All, 6]]]][[All, 1]]],
      NumberQ[#] &], 6, Method -> "Agglomerate"],
    FindClusters[
     Select[Flatten[
       Thread[Transpose[expars[[All, All, 5]]]][[All, 1]]],
      NumberQ[#] &], 6, Method -> "Agglomerate"],
    FindClusters[
     Select[Flatten[
       Thread[Transpose[expars[[All, All, 6]]]][[All, 1]]],
      NumberQ[#] &], 6, Method -> "Agglomerate"]
    }, BarOrigin -> Left, ChartLayout -> "Stacked",
  ChartLegends -> {"\!\(\*SubscriptBox[\(P\), \(8\)]\)E",
    "\!\(\*SubscriptBox[\(P\), \(9\)]\)E",
    "\!\(\*SubscriptBox[\(P\), \(8\)]\)EA–ER",
    "\!\(\*SubscriptBox[\(P\), \(9\)]\)EA–ER"}]
SetDirectory[thesisdir]
Export["parsestfig1.eps", parsestfig1]
Export["parsestfig2.eps", parsestfig2]
Export["parsestfig3.eps", parsestfig3]
Export["parsestfig4.pdf", parsestfig4]
(*Export all treat to .mx*)
in = AbsoluteTime[];
ParallelTable[
  tocsv[treat[obs[p, i]]], {i, 10, 99}, {p, pre[[2 ;; 4]]}];
ParallelTable[
  tocsv[treat[obs[p, i]]], {i, 10, 59}, {p,
   pre[[5 ;; Length[pre]]]}];
out = AbsoluteTime[];
Needs["ErrorBarPlots`"]
r1 = Show[{
   BarChart[{Mean[a50], Mean[a100], Mean[a125]},
    ChartLabels -> {50, 100, 125}],
   ErrorListPlot[
    Table[{Mean[i],
      StandardDeviation[i]/2}, {i, {a50, a100, a125}}]]},
  PlotLabel -> ""]
Export["r1.eps", r1]
```