COMPUTERISED SCORING AND GRAPHING OF THE FARNSWORTH-MUNSELL 100 HUE TEST OF COLOUR VISION - A PROGRAM WRITTEN IN PASCAL

V Balakrishnan, P T K Chew

ABSTRACT
The Farnsworth Munsell 100 Hue test is the best test for diagnosing and assessing colour vision defects. A computer program written in Pascal which automatically scores, plots and analyses the results of the Farnsworth Munsell 100 Hue test is presented.

Keywords: Colour vision, psychophysical tests, Farnsworth-Munsell, computer, Pascal

INTRODUCTION
Colour vision testing is an essential part of any comprehensive clinical assessment of visual function. Various clinical tests have been designed and are used to varying extents depending upon availability and the user's familiarity.

The Farnsworth Munsell 100 Hue test is presently the 'gold standard' against which other tests are measured. It was originally designed to test colour discrimination of employees in the dye and paint industries. However, with time, it has been found to be extremely useful in the detection, diagnosis, qualitative and quantitative assessment of both congenital and acquired colour vision deficiencies.

The test consists of 84 plastic caps with different coloured tops which systematically sample the full range of colours perceptible by the human eye. Each cap is just distinguishable from its neighbour based on its hue. The 84 caps form a sequence which a 'normal' would be able to arrange with few mistakes. 'Abnormal' persons would make characteristic errors because they would have difficulty discriminating certain hues. The test would therefore reveal both the orientation and degree of the defect. It also facilitates classification of 'normals' into superior, average or poor colour discriminators. Furthermore, it is the best method presently available for the detection of early acquired colour vision defects.

Unfortunately, the test is laborious and time consuming. Calculations are needed to score and plot the results for interpretation. The typical patient would require around 10 to 15 minutes per eye to perform the test. However, the calculations could easily take 30 minutes. This has resulted in the test falling into disuse except in specialised centres (with cheap trainee labour).

In Singapore, the most commonly used test of colour vision is the Ishihara pseudo-isochromatic plates which were designed for detecting significant congenital red-green defects. Unfortunately, it does not detect blue-yellow defects and is not a quantitative test, thus precluding its use in following a patient's progress. As acquired colour vision defects often involve the blue-yellow axis, complete reliance on the Ishihara plates defects does involve some risk.

The easy availability of microcomputers suggested that the drudgery of the Farnsworth Munsell test could be reduced if a user-friendly and fast program was written. It was hoped that the savings of time would allow the test to be used on a wider basis and benefit patient care in so far as it allowed a more comprehensive assessment. This program is presently used by ophthalmic units in Singapore and Malaysia.

SUBJECT AND METHODS
The program was written in Turbo Pascal Version 5.5 and runs on any IBM compatible computer with a graphics adapter and an Epson compatible printer. This configuration allows it to be used on the majority of micro computer systems presently available.

The user is requested to input patient's name, age, date and eye being tested. Next the sequence of caps which the patient arranged is input. Thereafter, everything else is automated.

The following results are output:
- Total error score
- Red-green partial error score
- Blue-yellow partial error score
- Individual cap error scores

The statistical significance of the scores are also computed based on age matched norms established by Verriest et al. The colour axis determination is made using the Smith's method.

A polar graph is also plotted on screen which facilitates identification of the orientation of the colour deficiency. (See Fig 1 which shows a patient with a typical red-green defect).

The scores, statistical analysis and graph may all be printed for filing purposes. The program also stores the results for future reference and comparison.

DISCUSSION
There have been several attempts at automating the Farnsworth Munsell test in the past. Benzschawel published a program written for the now obsolete Apple computer which calculated error scores and compared them to population norms. However, it did not plot the polar graph which has traditionally been used for interpretation.

Another program for the Apple computer was written in 1988 by us which addressed this deficiency

Miguel Lugo et al. published a program in BASIC for the IBM PC which calculated error scores and plotted a polar graph. This program did not perform statistical analyses or calculate separate red-green and blue-yellow scores.

Our present program is the only one published written in Pascal which performs all the functions listed above. Its advantages are:
We have found it invaluable in our daily clinical practice.

The program listing from page 156 to 159 contains an abridged version of the program. It is fully functional, but obviously lacks the whistles and bells of the full program which is too long for publication in this journal. The full program can be obtained from the author on request.

FUTURE DEVELOPMENTS
We are presently working on a totally computerised anomaloscope which will enable quick accurate testing of colour vision without the need to purchase a separate anomaloscope. This should provide anyone with access to an IBM microcomputer (with a VGA monitor) with an immediately useful clinical tool for everyday use.

REFERENCES
program Farnsworth;

uses Crt,Dos,Graph;

(*Farnsworth,Munsell 100 Hue scoring and graphing*)

{Short Version 7}

(*Copyright 1990, Dr Vivian Balakrishnan, NUS, Singapore*)

type
  info = record
    age: integer;
    which_eye: char;
    c: array[1..84] of integer;
  end;

Int84arraytype = array[1..84] of integer;

var
  c,CapScore: Int84arraytype;

(*Raw cap arrangement,ie, c[1] is the numerical label of*
  *the 1st cap , c[84] is numerical label of the last cap*)

Total_Error_Score,BY_Score,RG_Score: integer;

n,BoxNo : shortint; ch,pr : char; testdata : info;

procedure initial_assignment;
begin
  testdata.age := 30;
  testdata.which_eye := 'r';
  for n := 1 to 84 do c[n] := n;
end;

procedure display_caps;
var n : byte;
begin
  clrscr;
  for n := 1 to 84 do
    begin
      GotoXY(((n-1) mod 21)+1)*3, (((n-1) div 21)+1)*3);
      write(c[n]);
    end;
end;

procedure getinput;
var pos,EmptyCapPos,n : byte; displaced_cap,inputinteger,Code : integer;
inputstring: string[2];

begin
  pos := 1; displaced_cap := 0;
  repeat
    GotoXY(((pos-1) mod 21)+1)*3, (((pos-1) div 21)+1)*3);
    TextAttr := 16; write(c[pos]); NormVideo; ch := readkey;
    if ch = #0 then ch := readkey;
    case ch of
      #77 : inc(pos,1); #75 : dec(pos,1); #80 : inc(pos,21); #72 : dec(pos,21);
      (*numerical key*)
      '0'..'9' : begin
        inputstring[1] := ch;
        repeat ch := readkey; until (ch >= '0') and (ch<='9');
        inputstring[2] := ch; inputstring[0] := #2;
        val(inputstring,inputinteger,Code); n := 1;
        while c[n] <> inputinteger do inc(n); EmptyCapPos := n;
        if pos > EmptyCapPos then
          for n := EmptyCapPos to (pos-1) do c[n] := c[n+1];
        else
          for n := EmptyCapPos downto (pos+1) do c[n] := c[n-1];
        c[pos] := inputinteger;
      end;
    end;

    if pos < 1 then pos := 1; if pos > 84 then pos := 84; display_caps;
    until (ch = #13) and (displaced_cap = 0) or (x < 3) or (x > 66) or (y < 3) or (y > 12));
end;

procedure CalculateScores;
var n : byte;

begin
  Total_Error_Score := 0; BY_Score := 0; RG_Score := 0;
  for n := 1 to 84 do
    begin
      case n of
        (adjacent to left anchor cap)
          1,22,45,64 : CapScore[c[n]] := Abs(c[n]-(n-1)) + Abs(c[n]-c[n+1]);
        (adjacent to right anchor cap)
          21,42,63,84 : CapScore[c[n]] := Abs(c[n]-(n+1)) + Abs(c[n]-c[n-1]);
        else
          CapScore[c[n]] := Abs(c[n]-(n-1)) + Abs(c[n]-c[n+1]);
      end;
    end;
end;
end; (case)
Total_Error_Score := Total_Error_Score + CapScore[c[n]] - 2;
case c[n] of
1..12, 34..54, 76..84 : BY_Score := CapScore[c[n]] - 2 + BY_Score;
13..33, 55..75 : RG_Score := CapScore[c[n]] - 2 + RG_Score;
end; (case)
end;
end;
end;
function critical_score(age_of_testee: integer; which_eye: char): real;
type scoretype = array[1..14] of real;
const
age: scoretype = (12, 16, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75);
monoscore: scoretype = (193, 135, 110, 104, 110, 124, 142, 164, 189, 213, 234, 256, 281, 317);
binoscore: scoretype = (154, 114, 93, 89, 106, 123, 137, 151, 161, 172, 178, 188, 200, 216);
var
score: scoretype;
begin
case which_eye of
'B', 'b': score := binoscore; else
score := monoscore;
end;
case age_of_testee of
0..12: critical_score := score[1];
75..200: critical_score := score[14];
else
n := 1;
while age_of_testee >= age [n] do inc(n);
critical_score := score[n] + ((score[n] - score[n - 1]) / (age[n] - age[n - 1])) * (age_of_testee - age[n - 1]);
end;
end;
end;
procedure InterEyeComparison;
var score1, score2, C: real;
begin
ClrScr;
Write('What is the total error score of the first eye? ');
Readln(score1);
Write('What is the total error score of the second eye? ');
Readln(score2);
GotoXY(10, 12); C := Abs((Sqrt(score1) - 0.25) - Sqrt(score2));
if C > 2.39 then
begin
Write('Difference between eyes is significant (p < 0.05)');
end
else Write('Difference between eyes is not significant (p > 0.05)');
Readln;
end;
function IntToStr(i: integer): string;
var s: string[3];
begin
Str(i: 3, s); IntToStr := s;
end;
procedure DisplayResults;
var
CapArrangement, RowScore: array[1..4] of string[63];
n, m: byte; f: text;
begin
assignCrt(f);
Rewrite(f); ClrScr;
write(f, 'Age: ', testdata.age);
write(f, 'Test Data: ', testdata.which_eye, ' eye');
write(f, 'Total Error Score is ', Total_Error_Score);
write(f, 'Partial Blue-Yellow error score is ', BY_Score);
write(f, 'Partial Red-Green error score is ', RG_Score);
if Total_Error_Score > Critical_Score(testdata.age, testdata.which_eye) then
write(f, 'Abnormal result - p < 0.05');
else
write(f, 'Within normal limits - p > 0.05');
case testdata.which_eye of
'B', 'b': write(f, 'Critical score for binocular');
else
write(f, 'Critical score for monocular');
end;
write(f, 'Test for subject aged ', testdata.age, ' is');
WriteLn(f, Critical_Score(testdata.age, testdata.which_eye): 3: 1);
procedure StartUpGraphics;
var Gd,Gm : integer;
begin Gd := Detect; InitGraph(Gd,Gm,''); end;

procedure PlotGraph(Radius : Int84arraytype; Xasp,Yasp : word);
var m,n,FirstX,FirstY,PriorX,PriorY,CircleBaseRadius,CircleRadiusStep : integer;
p : word; ArcCoords : ArcCoordsType; st : string[2];
begin
SetAspectRatio(Xasp,Yasp); CircleBaseRadius := 10; CircleRadiusStep := 10;
SetTextJustify(CenterText,CenterText); SetTextStyle(SmallFont,HorizDir,2);
OutTextXY(75,(GetMaxY-5),'Copyright 1990, Vivian Balakrishnan');
for m := 1 to 10 do
for n := 1 to 42 do
begin
p := round(((n-1)*360*2/84)+90);
Arc(((GetMaxX+1) div 2),((GetMaxY+1) div 2),p,p+1,(m*2)*CircleRadiusStep+CircleBaseRadius);
if m = 10 then
begin
Str((n*2)-1,st); GetArcCoords(ArcCoords);
SetTextStyle(SmallFont,HorizDir,6);
OutTextXY(ArcCoords.Xstart,ArcCoords.Ystart,st);
end
else if (n = 1) or (n = 22) then
begin
Str(m*2,st); GetArcCoords(ArcCoords);
SetTextStyle(SmallFont,HorizDir,4);
OutTextXY(ArcCoords.Xstart,ArcCoords.Ystart,st);
end;
end;
for m := 1 to 84 do
begin
p := round(((m-1)*360/84)+90);
Arc(((GetMaxX+1) div 2),((GetMaxY+1) div 2),p,p+1,(m)*CircleRadiusStep+CircleBaseRadius);
GetArcCoords(ArcCoords);
with ArcCoords do
begin
case m of
1 : begin FirstX := Xstart; FirstY := Ystart; end;
84 : begin
Line(Xstart,Ystart,PriorX,PriorY); Line(Xend,Yend,FirstX,FirstY);
end;
else Line(Xstart,Ystart,PriorX,PriorY);
end; (case)
PriorX := Xend; PriorY := Yend;
end; (with ArcCoords)
end; (for m := 1 to 84)
end;
end;

begin
end;

procedure GetBioData;
var
age : string[3]; which_eye_str : string[1]; which_eye : char; Code : integer;
begin
writeln;
write('Age of patient '+IntloStr(testdata.age));
GotoXY(36,WhereY); readln(age);
va

if byte(which_eye_str[0]) = 1 then testdata.which_eye := char(which_eye_str[1]);
end;

begin
repeat
TextMode(LastMode); Clrscr; GotoXY(10,3);
Write('Farnworth-Munsell 100 Hue Test Analyser');
GotoXY(10,10); Write('1) Run a new test');
GotoXY(10,14); Write('3) Inter-eye comparison');
GotoXY(10,18); Write('5) Quit');
GotoXY(10,20); Write('Please type the number of your choice');
Readln(ch); Clrscr;

end;
REACHING NEW HEIGHTS IN CUSTOMER SATISFACTION

It is 8:30am, Monday morning in Johnson & Johnson's office. Everyone is in the conference room. In today's monthly meeting, a number of important issues are discussed. Almost all issues discussed are related to customer satisfaction. "How can we serve our customers better?" "Should we try this way from now on?" At Johnson & Johnson, everyone is talking about our most important mission - TOTAL CUSTOMER SATISFACTION.

At 11:45am, Caroline and Susan went out to lunch. A little too early? No, we have implemented a flexible lunch schedule so that we have full telephone coverage throughout lunch for customer service function. At 12:30, it was Selvi and Elida's turn for lunch. This is just one example of how Johnson & Johnson is introducing new policies to serve its customers better.

In 1990, we conducted a nation wide customer satisfaction survey. We asked you what you value most in terms of services offered by suppliers and qualities of sales representatives. We, then, asked you how we are doing against other suppliers on these attributes. Based on your valuable inputs, we have now identified what services you require of us. You told us we were doing better than other suppliers on most attributes. Although we were happy with your performance rating on Johnson & Johnson, we do not want to be complacent.

Reaching New Heights is our theme for 1991. From top management down the line, we have committed ourselves to strive for TOTAL CUSTOMER SATISFACTION. Based on the 1990 survey results, we have set goals and plans to improve our customer satisfaction level. Our ultimate goal is improved health care for Singapore through providing the highest quality products and services to our valued customers.

SERVING HOSPITALS AND MEDICAL PROFESSIONALS

Johnson & Johnson

ETHICON

Codman

Johnson & Johnson PATIENT CARE

CRITIKON

JOHNSON-JOHNSON ORTHOPAEDICS

SURGIKOS

VISTAKON

3 INTERNATIONAL ROAD, JURONG, SINGAPORE 2261. TEL: 2658922, TELEX: JANUSIN RS24868