

תיעוד השמשת מערכת Eyelink-II לצורך ניסויים

1. הקדמה

מסמך זה נועד לעזור להשמיש מחשב PC כמחשב ניסויים למערכת Eyelink-II. ההנחה הגלומה היא שההתקנה הפיזית של המערכת הושלמה, כלומר ישנו מחשב שרת (Host PC), המכיל כרטיס PCI יעודי, אליו מחובר מתקן העקיבה. כמו כן ישנו מחשב לקוח (Display PC), המשמש לניסויים ומחובר בכרטיס רשת וכבל מוצלב (תצורת Back-to-Back) למחשב השרת והרשת ביניהם נבדקה ועובדת. כמו כן נניח שמערכת ההפעלה היעודית ROMDOS הותקנה במחשב השרת. אם לא כך הדבר, יש לפנות למדריך ההתקנה **EyeLink II Installation Guide** ולסיום, המסמך מניח שלמתקין יש סמכויות אדמיניסטרטור למחשב ה-PC בו הוא עובד.

2. הסבר כוללני על המערכת

המערכת מורכבת מאופן כללי משלושה חלקים:

1. עוקב העיניים – המתקן שמתלבש על ראש הנבדק ומתעד את תנועות אישוניו
2. מחשב השרת - מחשב המכיל חומרה ותוכנה יעודית שמתעד את המידע מהעוקב.
3. מחשב התצוגה – מחשב זה מנהל את הניסוי מכל ההיבטים.

בניסוי ממוצע התהליך הוא כזה:

הנבדק יושב מול מחשב התצוגה ומפעיל את תוכנת הניסוי (Picture.exe). תוכנת הניסוי מפעילה, דרך הרשת, את מחשב השרת ודרכו את עוקב העיניים, כדי שהמידע יתחיל להיות מתועד באופן מקומי בקובץ בעל סיומת edf ובעל פורמט ייחודי. בשלב הזה התוכנה מתחילה להראות לנבדק תמונות שונות, בהפרשי זמן שונים. עם תום הניסוי התוכנה מפסיקה את פעולת השרת והעוקב, ומעתיקה את המידע הנקלט בשרת לדיסק המקומי של מחשב התצוגה. לאחר מכן ניתנת אפשרות, באמצעות תוכנה אחרת, לבחון את יומן הניסוי, השמור כקובץ edf, באמצעות תוכנת Data Viewer.

3. תוכנות דרושות

לצורך יצירת תשתית ניסויית יש צורך במספר תוכנות שיותקנו על מחשב הלקוח.

1. דיסק ה- Eyelink II – דיסק זה מכיל לא רק את הספריות הדינמיות (DLL) הדרושות לתיקשור עם העוקב אלא גם דוגמאות מוכנות מראש. יש להכניס את הדיסק לכונן, להיכנס לספרייה Display PC installer\Windows ולהפעיל את Eyelink display Software
2. דיסק Data Viewer – תוכנת ה- Data Viewer משמשת לבחינת קובץ ה- EDF הנוצר כתוצאה מניסוי ספציפי. יש להכניס את הדיסק לכונן ולהפעיל את EyelinkDV_1.3.60
3. תוכנת ה- Key - תוכנת ה- Data Viewer מוגנת בידי מנגנון הגנה בצורת פלג USB. ישנם שני סוגים. הסוג בו נתקלנו הוא HASP שהוא פלג אבטחה סטנדרטי של חברת אלדין. למרות שהוא מוגדר כ- Plug-n-Play יש לפני כן להתקין את תוכנת ה- USB, שהותקנה לדיסק יחד עם תוכנת ה- Data Viewer. שם התוכנה הוא HDD32.exe והיא בדר"כ שוכנת בספרייה "C:\Program Files\SR Research\Eyelink\DataViewer\EyeLink Data Viewer"

4. בדיקה בסיסית

כדי לבדוק שהתוכנות הותקנו נכון (בהנחה שההתקנה הפיזית נבדקה) ניתן לעשות מספר מבדקים:
Eyelink II – הבדיקה הפשוטה והיעילה ביותר היא קובץ exc בשם track המאפשר הרצת "מיני-ניסוי", כולל בדיקת איפוס, כיוון והצגת שלוש תמונות.
Data Viewer – כדי לבדוק שהתוכנה הותקנה יש פשוט להריץ אותה על קובץ edf קיים. עדיף על זה שה tracker ייצר. ניתן להתעלם לצורך העניין מההודעה על רשיון חסר.
USB key – כדי לבדוק שהתוכנה הותקנה היטב יש לבדוק ב "Start -> Control Panel -> Add or Remove Programs" שם אמורה להופיע תוכנה בשם "HASP Device Driver". אישור סופי לתקינות ההתקנה היא הפעלה של Data Viewer עצמה ויודוא שאין הודעה המספרת על "רשיון זמני" (Demo License).
אם עדיין ישנה בעיה, יש להוריד את התוכנה <http://69.159.205.54:8000/download/Hinstall.zip> ולהפעילה

5. מבנה קובץ קונפיגורציה

N

<display time at milisec for picture #1>:<file name for picture #1>
<display time at milisec for picture #2>:<file name for picture #2>

<display time at milisec for picture #N>:<file name for picture #N>

כש N הוא מספר התמונות שיוצגו לנבדק

6. מבנה הקוד

זהו החלק העיקרי במסמך, חלק זה מתאר את השינויים שנעשו בקוד C קיים (Template) כדי להפכו לקוד דינאמי. לצורך בחינה מעמיקה של הקוד, בפרט חלקים שלא שונו. מומלץ לעיין במדריך למתכנת:

Programming EyeLink® Experiments in Windows Version 2.1 (SDL)

הקוד שממנו קומפלה התוכנית מורכב ממספר קבצים, שנלקחו ממה שקרוי בתיעוד "Picture Template"

– main.c

קוד זה הוא הקוד הראשי. הוא מטפל בקריאה לתוכנית מתוך windows או מדוס (אם הסביבה מתאימה).

1. התוכנית הראשית בודקת את הארגומנטים הניתנים לה ב command line ע"י הפונקציה `parseArgs`. לקוד הפונקציה הזו נוסף ארגומנט נוסף בו היא יכולה לטפל "-cfg", שלאחריו רשום קובץ הקונפיגורציה.
3. לאחר בדיקת הארגומנטים התוכנית קוראת את קובץ הקונפיגורציה (במידה והוון) ע"י הפונקציה `readCfgFile()`.
4. בזמן טעינת הקובץ התוכנית מקצה באופן דינאמי מקום לטעינת התמונות ומאחסנת את המידע בו.
5. לאחר מכן מופעלת הפונקציה הראשית `app_main()`, פונקציה זו מאתחלת את חלון התצוגה, יוצרת את קובץ ה edf בו ישמרו תוצאות הניסוי, מאתחלת את הקשר לעוקב העיניים ובסופו של דבר מפעילה את הפונקציה `run_trials()`, שמתחילה את הניסויים עצמם.

6. הפונקציה נמצאת בקובץ אחר "trails.c" ולכן כל המידע הרלוונטי מועבר לה:

- a. NTRIALS – מספר הניסיונות המתבצעים (מספר התמונות שהולכות להיות מוצגות)
- b. images, imgname - מערך המכיל את שמות התמונות שהולכות להיות מוצגות
- c. waitBetweenImgs – מערך המכיל את הזמן בו יש להציג כל תמונה

– trails.c

קוד זה הוא הקוד האחראי לפעולות הנעשות לפני ואחרי כל ניסיון (הצגת תמונה), בינהן טעינת התמונה והתחלת התיעוד.

1. הפונקציה הראשית פה היא, כאמור, `run_trials()`
2. הפונקציה רצה בלולאה על כל התמונות במערך `images` ועבור כל תמונה, בהנחה שהניסוי לא הופסק, היא מפעילה את `do_picture_trial()`
3. הפונקציה `do_picture_trial()` אחראית ל: טעינת התמונה למסך, תיעוד טעינת התמונה בקובץ ה edf, התחלת תיעוד תגובות הנבדק, למשך זמן הקבוע מראש, ע"י הפונקציה `bitmap_recording_trial()` מקובץ אחר "trail.c"

- trail.c

הקוד בקובץ זה הוא הקוד שמורץ, הלכה למעשה, בכל ניסיון וניסיון, ותפקידו להתחיל ולסיים את התיעוד באופן תקין.

1. הפונקציה הראשית פה היא `bitmap_recording_trial()`, המקבלת מצביע פנימי לתמונה ואת הזמן להצגתה, בתום ההצגה התמונה מוסרת, והמידע שנקלט מהנבדק (ע"י עוקב העיניים) נשמר לקובץ ה edf.
2. לפונקציה תפקיד נוסף, שאינו בקוד המקורי, והוא בדיקת הקשת מקשים מצד המשתמש, בזמן שהתמונה מוצגת והמידע מהעוקב נרשם.
3. המידע נקלט באמצעות פונקציה `getkey()` של windows, ורק אם המקש שהוקש הוא ספציפי (108 נכון לעכשיו או "j"), אזי הפונקציה תרשום את התיעוד להקשתו בקובץ ה edf באמצעות הפונקציה `eyemsg_printf()`
4. יש לשים לב, שכפי שהודגש קודם, לב העניין הוא בפונקציה ה `getkey()`, שכאמור אינה פונקציה Realtime ופועלת בכל מקרה, ולכן יכולה לגרום להאטת את התוכנית.

7. על ההבדל בין ניסוי דינמי לסטטי

לפני שניגשים לבצע ניסויים בהיקף נרחב יש לדעת את הצורה בה ניסויים היו אמורים להתבצע במערכת לעומת הדרך בהם מתבצעים באמצעות התוכנה Picture.exe. ע"פ הרשום במסמכי התיעוד, כל ניסוי אמור להיות מוכן מראש, כלומר על החוקר לחשוב על ניסוי, למצוא כיצד יש לשנות את הקוד הקיים, המשתמש בספריות הגישה השונות, לקמפל אותו ואז להריץ אותו. הניסוי בצורה הזו הוא סטטי, כלומר לא משתנה, אותן תמונות תיטענה בכל פעם שנריץ אותו, הקלט היחידי לניסוי הינו מעוקב העיניים, למרות שהפלט

הניסוי דרך התוכנה הינו ניסוי דינמי, כלומר התוכנה קוראת מקובץ קונפיגורציה, בזמן עלייתה, שם רשומים לה כל התמונות שהיא צריכה להראות בפני הנבדק. כמו כן התוכנה מקשיבה להקשות המקלדת מצד המשתמש, בנוסף לקלט הבא מהעוקב. כל האמור לעיל דורש, אם כן, זהירות יתרה בבואנו לתכנן ניסוי דינמי שכזה, יש לשים לב שיותר מדי תמונות או הקשות רבות מדי, עלולות לגרום לשגיאה בדגימת הנתונים. גודל השגיאה ומספר התמונות הקריטי אינו ידוע, יש לברר זאת בעתיד.