**מוצג פוטוגרמטריה - תוצאות בדיקות**

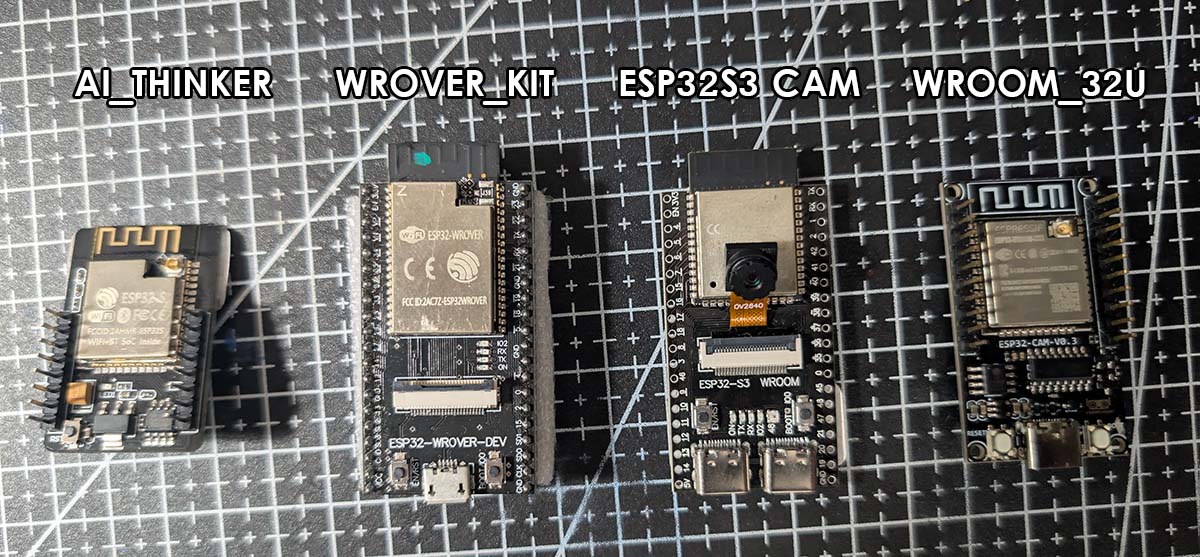
**From:** Ido Kronik <ido.kronik@mail.huji.ac.il>  
**Sent:** Saturday, August 30, 2025 8:14 PM  
**To:** Amir Ben. Shalom‏ <amirb@mada.org.il>  
**Subject:** מוצג פוטוגרמטריה - תוצאות בדיקות

היי אמיר מה שלומך?

הבדיקות על המיקרו-בקרים הסתבכו והיו מאתגרות יותר משציפיתי, לכן התעכבתי עליהן יותר מהצפוי. אבל עכשיו סיימתי איתן ואלו העדכונים:

קודם כל לגבי בדיקת איכות הקישוריות עם ובלי אנטנה חיצונית – מבין ארבעת סוגי הבקרים שבדקנו, רק לאחד יש חיבור לאנטנה חיצונית, וזה גם הבקר היחידי שאין לו בכלל זכרון PSRAM, מה שאומר שמלכתחילה הוא לא רלוונטי.  
בנוסף ביצעתי בדיקות על בקרי ה-ESP32S3 החדשים שהבאת לי. הסתבכתי מאוד עם הקנפוג שלהם ועם הצריבה, גם בגלל שהקונפיגורציה של ה-PSRAM שלהם אחרת משאר הבקרים וגם בגלל שכל אחד מחיבורי ה-USB-C שלהם תומך בפרוטוקול תקשורת אחר, אחד UART  ואחד OTG, מה שסיבך עוד יותר את הגדרות הצריבה ואת העבודה עם Serial Monitor. זה גם הציף כל מיני בעיות אחרות שהיו בתכנה שכתבתי, ושיניתי אותה בהתאם.

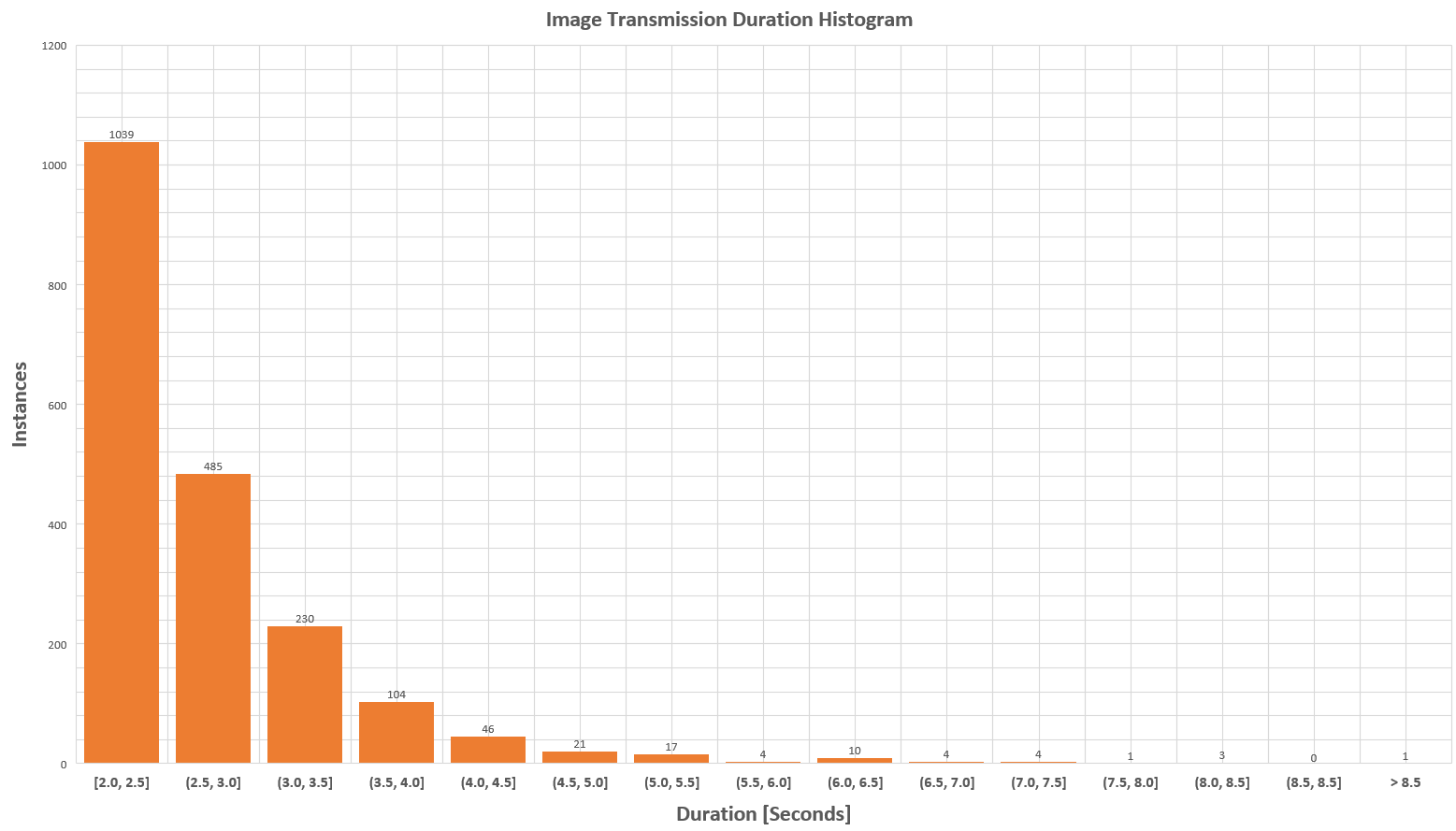
ביצעתי למעלה מ-1000 מדידות של לכידת תמונות אלחוטית באמצעות המערכת שבניתי, עם כל סוגי הבקרים. מצרף מדידות של RSSI ושל משך שידור התמונה הממוצעים של כל בקר. ניתן לראות שבקרי ה-ESP323S אכן הגיעו לתוצאות טובות יותר משאר הבקרים אך לדעתי לא בפער מספיק משמעותי כדי להצדיק את המחיר ואת מורכבות הקנפוג. לדעתי הבקר הכי מתאים לעבודה הוא ה-WROVER\_KIT. הוא לא הגיע אליי עם מדבקת הזמנה, מצרף תמונה שלו. מצרף גם טבלה שמסכמת את ההבדלים בין כל הבקרים שנבדקו.



A screenshot of a computer

AI-generated content may be incorrect.

עוד נתון שמדדתי הוא שהגדלת כמות הבקרים המחוברים לא השפיעה על משך שידור התמונה של כל בקר בנפרד. מה שאומר שפרק הזמן בו השרת מחכה שיחזרו אליו כל התמונות שווה למשך השידור הארוך ביותר של אחד מהבקרים. מצרף היסטוגרמה של כל זמני השידור (אינדיבידואלית לבקר) שמדדתי. אני מניח שבעקבות המסקנה לעיל, ניתן לייחס להיסטוגרמה משמעות הסתברותית למשך השידור של כל התמונות יחד.

  
עבור כל סוגי הבקרים:

A graph with numbers and a number of images

AI-generated content may be incorrect.ועבור בקרי WROVER\_KIT בלבד:

פיצ'ר חדש שהוספתי למערכת הוא עדכונים Over the Air – כלומר לאחר הצריבה הראשונית, כל בקר יודע לבדוק מול השרת אם קיימת גרסת קושחה עדכנית יותר, ובמידה שנמצאת הוא יודע להוריד אותה ולצרוב אותה על עצמו. לדעתי יכול מאוד להקל את הצריבה על מספר רב של בקרים לאחר כל עדכון.

ביצעתי גם בדיקות עם ה-Meshroom – הזמן שלוקח לו לייצר מודל כתלות במספר התמונות וברזולוציית כל תמונה.

אחת המסקנות שהבנתי היא שכדי ש-meshroom יצליח לחבר את כל התמונות צריך לרווח אותן במרווחים של כ-**15 מעלות**. מה שאומר שעבור צילום ב-360 מעלות נדרשות **24 מצלמות**. אבל שמתי לב שהריווח ב-15 מעלות הכרחי רק כדי לשמור על רציפות, ולמעשה אפשר להשאיר רווח גדול יותר מאחורי הראש של המצולם ובגלל כמות הפרטים הפחותה, התכנה תצליח לייצר תוצאה מספקת. נגיד, אם משתמשים רק ב-**18 מצלמות שמרווחות ב-15 מעלות**, וה-"חור" ממוקם מאחורי הראש של המצולם. במקרה כזה הצלחתי להגיע לתוצאות יפות גם ברזולוציית 800p ואפילו 400p **בזמנים של 47 שניות ו-22 שניות**, בהתאמה.  
מצרף גרף של זמן ייצור המודל כפונקציה של כמות התמונות, עבור 3 רזולוציות שונות. לא נגעתי בפרמטרים אחרים של התכנה מעבר לזה. ניתן לראות שההתנהגות היא יחסית לינארית בכל המקרים.

A graph of a graph with numbers and lines

AI-generated content may be incorrect.