

סרטוני AI עם Opus 5.5: הפרומפט הוא רק 10% מהתוצאה

מי שמנסה "פרומפט אחד וסרטון מוכן" מקבל סרטון גנרי. הפרומפט הוא רק 10% מהתוצאה: חוקי תנועה, מותג אמיתי, רפרנסים, סאונד ולולאת ביקורת עושים את השאר.

2026 באוקטובר · NexAi - Gilad

הגרסה המעודכנת באתר: nex-ai.co/he/guides/opus-5-5-ai-video-harness-guide/



Opus 5.5 לא מייצר קובץ וידאו בעצמו. הוא כותב קוד שמרנדר אותו, ולכן איכות הסרטון תלויה בסביבה שסביב הפרומפט: חוקי אולפן, מותג אמיתי, רפרנס, רשימת מפרט ולולאת ביקורת. הפרומפט עצמו אחראי לכ-10% מהתוצאה.

מאז ש-Opus 5.5 יצא, ברשת מסתובבים סרטונים שקשה להאמין שנוצרו על ידי AI: סרטוני השקה למוצרים, אנימציות ממשק, סצנות מפורטות. הרבה מהם מגיעים עם כיתוב כמו "לקח לי 15 דקות" או "פרומפט אחד בלבד". מי שמנסה את זה בבית מקבל בדרך כלל סרטון גנרי: טקסט באמצע המסך, רקע גרדיאנט, שום דבר שאפשר לזכור.

ניסיתי את זה בעצמי, ומה שהבנתי הוא שהפרומפט אחראי לכ-10% מהתוצאה. ה-90% האחרים הם הסביבה שמסביבו, מה שנקרא היום [harness engineering](#): חוקים, רפרנסים, קבצי סגנון ולולאת ביקורת. במדריך הזה אני עובר על מה שבניתי סביב הפרומפט ועל מה שהשתנה בגללו.

איך מודל שלא מייצר MP4 מייצר סרטון

מודל Opus 5.5 מקבל טקסט ותמונות ומחזיר טקסט. הוא לא יכול ליצור קובץ MP4 בעצמו. אני כותב לו בריף, טקסט ולצדו רפרנסים כמו תמונות, קישורים או סרטונים, והוא כותב קוד שמפיק את הסרטון. בדרך כלל הוא בוחר אחד מארבעה כיוונים:

- מצייר בקוד בעצמו, למשל עם SVG בדפדפן Chrome שרץ ללא ממשק.
- משתמש בפריימוורק כמו [Remotion](#), [HyperFrames](#) או [Diffusion Studio](#).
- מריץ צינור מעורב: מודלים של תמונה ושל וידאו, ומעליהם הרכבה בקוד.
- עובד על סרטון קיים: חיתוך, כתוביות, חומרי B-roll.

יש דוגמה ידועה של מי שהפיק סרטון בשוט אחד. כל הקוד ישב בקובץ `index.html` יחיד, הרינדור נעשה פריים אחר פריים בחלון ללא ממשק דרך [Playwright](#), ו-[FFmpeg](#) הרכיב את ה-MP4. הוא ציין שהמודל מעדיף לבנות הכול מאפס, בלי תלויות, ולא להשתמש ב-Remotion או ב-HyperFrames. מכאן המסקנה המעשית: אם לא מציינים פריימוורק, Opus 5.5 ילך לבד על הכיוון הראשון.

להתחיל: תיקייה, מודל ו-effort

פותחים את אפליקציית Claude לדסקטופ, יוצרים תיקייה חדשה כתיקיית עבודה, בוחרים Opus 5.5 ומגדירים effort גבוה. כדי לבדוק שהכול עובד נתתי לו פרומפט פשוט: סרטון של עשר שניות על תחזית מזג אוויר באפליקציה. השמש עולה, עננים מתקרבים, מתחיל גשם והטמפרטורה יורדת. התוצאה לא הייתה רעה, בדיוק מה שציפיתי. נתתי ל-Codex את אותו פרומפט והתוצאה שלו הייתה פחות מרשימה בהרבה.

אם HyperFrames או Remotion לא מחוברים אצלכם, ב-connectors אפשר לחפש HyperFrames, להתקין ולהתחבר לחשבון. עם Remotion עושים את אותו הדבר דרך התוסף שלו ל-Claude Code. מי שעובד עם Codex ולא עם Claude יכול להמשיך [למדריך עריכת סרטונים עם ChatGPT Codex ו-Remotion](#).

חוקי האולפן

הדבר הראשון שהייתי עושה בכל פרויקט הוא להדביק סט של "חוקי אולפן תנועה" בארבע קטגוריות:

- חוזה הרינדור, שעוזר לרנדור את הסרטון בצורה יציבה.
- המראה, שאוסר על סגנונות גנריים.
- סאונד.
- לולאת משוב.

אחרי שהדבקתי את החוקים הרצתי פרומפט של שורה אחת: סרטון תנועה דינמי של 15 שניות שמראה איזה מעצב תנועה מעולה אני, בתור רילס לקורות חיים, ושיתן את כל מה שיש לו. הוא עובד בגלל המבנה:

- "רילס לקורות חיים" קובע ז'אנר עם חוקים מוכרים.
- "איזה מעצב תנועה מעולה אני" הופך את המודל לנושא הסרטון, אז הוא מראה טכניקות במקום להסביר מוצר.
- 15 שניות קצר מספיק כדי לסיים במעבר אחד, וארוך מספיק לשישה עד שמונה שוטים.

- "תן את כל מה שיש לך" מכפיל את המאמץ שכבר הוגדר ב-effort.

יצאו תנועות מעניינות, הרבה מעברים וכמה כיווני עיצוב שונים, אבל הכול בשחור וכתום. זה לא מקרי: אלה הצבעים של Claude עצמו.

להכניס מותג אמיתי

הצעד הבא הוא להחליף את הפלט הגנרי במותג אמיתי, שלי או של לקוח. הרצתי את אותו פרומפט, הפעם "על Magic Path". יצא ממשק בתצוגת קנבס אינסופי, במצב כהה ובמצב בהיר, וסיום עם אנימציות לוגו. הסביבה נשארה אותה סביבה, והאנימציה והמיתוג התחלפו.

מרשים יותר לבקש מ-Claude לאסוף בעצמו את חומרי המותג: להשתמש בצילומי מסך אמיתיים של המוצר ובקבצי לוגו. הוא עושה את המחקר ואז מפיק סרטון הסבר מלא: מסביר את המוצר, מראה רכיבי ממשק, שולף צילומי מסך מהאתר ומייצר חלק מהאנימציות מאפס. כל מה שנשאר לי היה להחליף את ה-placeholders בשם המותג.

ניסיתי גם גרסה קצרה, "סרטון השקה של 15 שניות ל-Magic Path" ועוד שני משפטי הקשר, וקיבלתי סרטון נקי באותה מידה, בסגנון דומה, עם כל מה שצריך להראות.

להוסיף קול

אפשר להוסיף גם אודיו, אדם שמדבר בזמן שהמוצר על המסך. רוב הקרייניות של AI נשמעות שטוחות וגנריות, וחלק מה-APIים הטובים גובים לפי תו, כך שהחשבון מתנפח מהר. מה שעובד טוב: לחבר שירות text-to-speech דרך MCP כך שסוכן הקוד ישתמש בו ישירות. מגדירים את החיבור, קובעים הרשאות כלים ל-always allow ומבקשים משהו כמו "צור קריינות להוק פתיחה לסרטון יוטיוב על סוכני קוד". מקבלים כמה אפשרויות קול, אפשר לשבט את הקול שלכם ולהוסיף רגש, והיצירה לוקחת כמה שניות.

לעבוד עם רפרנסים

Opus 5.5 עושה עבודה מצוינת כשנותנים לו משהו לבנות עליו. האתר whatships.com הוא אוסף מסודר של סרטוני השקה והדגמות של סטארטאפים, וכל אחד מהם מקושר לפוסט המקורי ב-X. בוחרים סרטון שאוהבים, בגלל הסגנון, הצבעים, הפונטים או התנועה, והופכים אותו לרפרנס. הפרומפט שהשתמשתי בו, בניסוח שלי:

זה הרפרנס. הורד את הסרטון וחלץ פריים כל חצי שנייה. כתוב קובץ markdown של מדריך סגנון: פלטת צבעים, טיפוגרפיה, אורך שוטים, סוגי מעברים, תנועות מצלמה, גרעיניות, ואיך טקסט נכנס ויוצא. אחר כך כתוב רשימת שוטים קצרה לסרטון השקה של 15 שניות על Magic Path באותו סגנון, עם צילומי מסך אמיתיים. קח מהרפרנס את הדקדוק בלבד, אף פעם לא את התוכן, הלוגואים או הדמויות שלו.

יצא סרטון של המוצר שלי בסגנון הרפרנס. במקום ממשק של ציר זמן הוא הראה את הממשק האמיתי של המוצר.

עקביות: קבצי markdown ורשימת מפרט

כשרוצים שהתוצאות יהיו עקביות, כדאי לבקש מ-Claude לכתוב קבצי markdown שהוא חוזר אליהם. כך הוא זוכר את המותג, הסגנון, הדקדוק ואת הטון. אותו עיקרון עובד גם בבניית אתרים, כמו שכתבתי ב-7 שלבים לבניית אתר מקצועי עם [Codex](#). חוץ מזה צריך רשימת מפרט של רכיבים: כפתורים, מעגלי טעינה, סימני וי, גרפים ועוד.

בונים אותה מכמה קלטים:

- מה המוצר ומה צריך ממנו: לוגו, צבעי מותג, פונטים, צילומי ממשק.
- הכיוון, שמשנתה לפי סוג הסרטון. באנימציות רכיבי UI, למשל: "סרט מוצר, תנועת ממשק. מיכל אחד שלא נחתך אף פעם. כל מצב הוא אותו אלמנט שמשנה גודל, רדיוס ומילוי, בזמן שהתוכן מתחלף מאחורי טשטוש קצר." אפשר לבקש מ-Claude לעזור לנסח את זה לפי רפרנס.
- המבנה, כלומר איך בונים.
- ה-gotchas: מה המודל לא צריך לעשות ומה כדאי לו לזכור.

לפני שהוא כותב קוד, אני מבקש ממנו להציג את רשימת המצבים על רשת פעימות. זו טבלה שמתפקדת כשרטוט של האנימציה, פריים אחרי פריים. מתקנים כמה דברים אם צריך, ורק כשהטבלה מוצאת חן מרנדרים.

סאונד ורשת פעימות

יש טכניקה שבה כל תנועה מסונכרנת לצליל: שכבת מוזיקה, שכבת אפקטים ושכבת תמונה, כולן זזות יחד ב-BPM 120. הפרומפט, בניסוח שלי:

תן לסרט ציון לפי הקצב. סנתז בקוד רצועה מקורית ב-BPM 120, על אותו ציר זמן כמו התמונה. מדוד אותה והפק קובץ beats.json עם BPM, פעימות, פעימות ראשונות במשקל ואירועי הקשה. סנתז צלילי ממשק ב-Node: קליק על כל לחיצת סמן, וווש על כל מעבר של מיכל, ודפיקה עמוקה על הלוגו. אחר כך תזמן מחדש את האנימציה לפי הקובץ.

בגרסה הזאת יש הרבה יותר סאונד מבפעם הקודמת: צליל בכל פעם שטקסט קופץ למסך, וצליל בכל מעבר בין תמונות. אם זה טוב יותר או לא, תלוי בטעם. אבל זה ניסוי שכדאי לעשות.

בריק הבמאי

אחד מהשימושים שהכי אהבתי נעשה בפרומפט אחד. היוצר דיבר אל המחשב חמש דקות, Claude עבד 12 שעות, והוא התעורר לתוצאה. הפרומפט ארוך, אלפי תווים בכמה פסקאות, והוא מבוסס על סרטון מקורי שהיוצר עשה לו רימיקס. כשקוראים אותו רואים שהוא בנוי על שלד של 12 נקודות, כלומר בריק של במאי. אלה העיקריות:

- הסרט במשפט אחד.
- רפרנסים.
- כלים ומפתחות: אילו סקילים לטעון ואילו API-ים זמינים, למשל Seedance.
- ספר דמות לכל דמות: זוויות, פרופורציות, רגשות והבעות. אפשר לשמור בו מאות תמונות ומסמכים לכל דמות.

- גיליון שוטים (beat sheet).
- טקסט על המסך.
- שערי עבודה לפני כל מעבר לשלב הבא.
- לולאת ביקורת.
- תוצרים סופיים.

הסרט עורר אנשים אחרים לעשות לו רימיקס, וחלק מהתוצאות מרשימות מאוד, למשל אחת בסגנון אנימה רטרו.

יש גם תבנית פרומפט של במאי, והיא נפתחת כך: "אתה הבמאי, האנימטור, מעצב הסאונד ומהנדס הרינדור של סרט שנוצר בקוד". אחר כך היא עוברת על כל הנקודות שלמעלה. מעתיקים, מדביקים ומחליפים את הפרטים במה שבונים או במה שמשמש רפרנס. עם פרומפט ארוך ומובנה כזה התוצאה משתפרת. הממשק כולו נוצר, הטקסט מרחף פנימה, הסוכן מראה איך הממשק עובד בפועל מזווית מצלמה אחת, הטרמינל נפתח והעיצובים הופכים לקוד אמיתי.

לולאת הביקורת

בסוף, לולאת הביקורת: מבקשים מהמודל לנתח מה שעשה, לתת ציון לתוצאה, לתקן את הציונים הנמוכים ולחזור על זה עד שמגיעים לציון מסוים או גבוה ממנו. התוצאות הטובות ביותר אינן בניסיון אחד. אחד הסרטונים האלה לקח 163 קריאות למודל וכמעט שבע שעות. אני מעתיק את פרומפט הביקורת בכל פעם שיש תוצאה שאני רוצה לשפר: הוא עובר סבב אחרי סבב, משפר ציונים שונים וממשיך לשמור את כולם בשבע או שמונה ומעלה. את ההחלטה אילו תוצאות לשמור כדאי להשאיר לאדם, כמו שכתבתי על [טעם כיתרון תחרותי](#).

לארוז ולמכור

השלב האחרון הוא לארוז את התוצר: לעצב את הפורמט, או להפוך את כל התהליך לסקיל או לשירות. אפשר לארוז אותו כסקיל בשם מסוים. בפעם הבאה הוא ישאל אותי את המידע שהוא צריך, כמו כתובת האתר, משך הסרטון והפורמט, ואז יריץ את כל הצינור לפי הכללים הקשיחים. יש כבר מי שמוכר את זה כשירות: בסביבות 70 דולר להפקת סרטון כזה למות של מישהו אחר. לדעתי זה סוג שירות מעניין להציע היום.

שאלות נפוצות

האם Opus 5.5 יכול ליצור קובץ MP4 בעצמו?

לא. Opus 5.5 מקבל טקסט ותמונות ומחזיר טקסט. הוא כותב קוד, למשל SVG בדפדפן Chrome ללא ממשק, שימוש בפריימוורק כמו Remotion או HyperFrames, או צינור מעורב עם מודלי וידאו. כלי רינדור מפיק מהקוד את קובץ ה-MP4.

באיזה פריימוורק כדאי להשתמש?

אם לא מציינים פריימוורק, Opus 5.5 נוטה לבנות הכול בקובץ HTML אחד ולרנדור אותו עם Playwright ו-FFmpeg, בלי תלויות. כדי לעבוד עם Remotion, HyperFrames או Diffusion Studio צריך לבקש אותם במפורש בבריף.

כמה סבבים נדרשים עד שהסרטון טוב?

התוצאות הטובות ביותר אינן בניסיון אחד. אחד הסרטונים שתיארתי לקח 163 קריאות למודל וכמעט שבע שעות. לולאת ביקורת עם ציונים עוזרת לשמור את כל הרכיבים בשבע או שמונה ומעלה.

איך מוסיפים קריינות לסרטון?

מחברים שירות text-to-speech דרך MCP, כך שסוכן הקוד משתמש בו ישירות. מגדירים את החיבור, קובעים הרשאות כלים ל-always allow ומבקשים קריינות להוק פתיחה, למשל. חלק מה-APIים הטובים גובים לפי תו, ולכן כדאי לבדוק את התמחור לפני שמתחילים.

מאיפה מתחילים?

אם מתחילים מאפס, הייתי הולך בסדר הזה: חוקי אולפן, מותג אמיתי, רפרנס, ולבסוף לולאת ביקורת.

לעדכונים, להעתקת פרומפטים בלחיצה ולקישורים נוספים, חזרו למדריך באתר.