**Bilateral Filtering for Gray and Colored Images**

מבוא

Bilateral Filtering הוא אלגוריתם שמטרתו "להחליק" תמונות שחור-לבן או צבעוניות.

הרעיון הכללי של פעולת ההחלקה של התמונה הוא הוספת ממוצע משוקלל של מרחקי הפיקסלים השכנים לכל פיקסל בתמונה המקורית.

בשיטה זו, מחליפים את ערך הפיקסל בנקודה X בתמונה, עם הערך הממוצע שחושב באמצעות פונקציית הGaussian. ערך ממוצע זה מייצג את הדמיון הפוטומטרי בין הפיקסל בנקודה X הנ''ל לבין שאר הפיקסלים שנמצאים בטווח שבחרנו לבדוק והקרבה הגאומטרית של הנקודה X לשאר הפיקסלים בתחום.

מרחקי הפיקסלים כאמור, נמדדים בשני מימדים: מימד גיאומטרי ומימד פוטומטרי.

מרחק גיאומטרי- האלגוריתם יתחשב בכל איטרציה אך ורק בפיקסלים הנמצאים לא רחוק יותר ממרחק מוגדר מראש מהפיקסל באיטרציה הנוכחית.

מרחק פוטומטרי- האלגוריתם ייתן משקל לכל אחד מבין הפיקסלים השכנים (מבחינת מרחק הגאומטרי) לפי המרחק הפוטומטרי של הפיקסלים, כך שפיקסלים הקרובים בערכם לפיקסל באיטרציה הנוכחיות יקבלו משקל גבוה יותר מפיקסלים אשר רחוקים בערכם.

חישוב המשקלים מתבצע תוך שימוש בפונקציית ההתפלגות .Gaussian בהמשך יופיע פירוט בנוגע לפרמטרים בהם השתמשנו בפונקציה.

בתמונה צבעונית (RGB) כל פיקסל מוגדר על ידי שלושה ערכים: ערך אדום, ערך ירוק, ערך כחול. כאשר קלט האלגוריתם הוא תמונה צבעונית נריץ את האלגוריתם עבור כל אחד משלושת הצבעים המגדירים את הפיקסל בנפרד.

שיטת החישוב

**הנוסחה לחישוב ערך כל פיקסל באלגוריתם** Bilateral Filter **היא:**

לאחר נרמול:

כאשר:

* : הפלט המוחזר, כלומר תמונה שעברה החלקה.
* : הקלט, כלומר פיקסל בתמונה עליה נרצה להפעיל את הפילטר.
* : המרחק הגיאומטרי בין , נקודת המרכז, לפיקסל .
* : המרחק הפוטומטרי בין , נקודת המרכז, לפיקסל .
* : הפונקציה לאחר נרמול.

לצורך מימוש האלגוריתם בכלים תכנותיים השתמשנו בנוסחה הבאה לחישוב ערכו של כל פיקסל בתמונה:

כאשר:

* : קבוצת הפיקסלים שנמצאים בתחום החלון שהוגדר עבור פיקסל .
* : משקל הפיקסל השכן, , על פי המרחק הגיאומטרי שלו מ-. מחושב על ידי פונקציית Gaussian עם הפרמטר .
* : השונות בה נשתמש בחישוב משקל שכן q לפי מרחקו הגאומטרי מ-p תוך שימוש בפונקציית Gaussian.
* : משקל הפיקסל השכן, , על פי המרחק הפוטומטרי שלו מ-. מחושב על ידי פונקציית Gaussian עם הפרמטר .
* : השונות בה נשתמש בחישוב משקל שכן q לפי מרחקו הפוטומטרי מ-p תוך שימוש בפונקציית Gaussian.
* : ערך הפיקסל

כאמור, אלגוריתם Bilateral Filter פועל בשני מימדים- מרחק פוטומטרי ומרחק גיאומטרי בין הפיקסלים בתמונה.

השיטה בה האלגוריתם מבצע את החלקת התמונה היא על ידי החלפת ערכו של כל פיקסל בתמונה עם ממוצע משוקלל של ערכי הפיקסלים שנמצאים בקרבתו.

ראשית, האלגוריתם מגדיר תחום גאומטרי ריבועי, כך שהפיקסל אותו אנו מחשבים באיטרציה הנוכחית, P, נמצא במרכזו. נקרא לתחום זה "חלון" ולפיקסלים הנמצאים בתוך החלון "שכנים של P".

האלגוריתם משתמש בפונקציית ההתפלגות Gaussian לחישוב משקלו של כל פיקסל בחלון לפי מרחקו הגאומטרי והפוטומטרי מ-p.

בתמונות שחור-לבן כל פיקסל מיוצג על ידי ערך יחיד. ערך זה יהווה את בחישוב.

אולם, תמונות צבעוניות מיוצגות על ידי שלושה ערכים: ירוק, כחול, אדום.

עבור תמונות צבעוניות האלגוריתם יבצע את החישובים תוך התייחסות לכל אחד מהצבעים המגדירים פיקסל בנפרד. כלומר, ניצור שלוש תמונות נפרדות עבור כל אחד מהצבעים: אדום, ירוק וכחול. ערכו של כל פיקסל בתמונה האדומה יהיה הערך ה ב של הפיקסל בתמונה המקורית, ובהתאמה עבור ו- בתמונה הירוקה והכחולה. נריץ את האלגוריתם על כל אחת מתמונות אלה לקבלת תמונות מוחלקות. את התמונות המוחלקות נמזג לכדי תמונה צבעוניתמוחלקת.

דיון בתוצאות

**הפעלנו את האלגוריתם על תמונה מורעשת עבור ערכים משתנים של גודל החלון. מצאנו כי גודל החלון האידאלי הוא 7. האידאליות של גודל זה נובעת מזמן הריצה ואיכות התוצאות שהתקבלו. עבור ערכי גודל חלון קטנים יותר זמן הריצה הצטמצם, אך פחות שכנים נלקחו בחשבון בחישוב הממוצע ולכן איכות התמונה ירדה.**

**עבור ערכי גודל חלון הגדולים מ-7 זמן הריצה עלה משמעותית, ובנוסף משקל הפיקסלים הרחוקים היה קטן מאוד ולכן לא הבחנו בעלייה משמעותית באיכות התוצאות.**

נציג תמונה מורעשת ואת תוצאות הרצת האלגוריתם Bilateral Filtering על תמונה זו, תוך שימוש בפרמטרים משתנים וגודל חלון קבוע בעל הערך 7.

התמונה המקורית:



תוצאות הרצת האלגוריתם עבור גודל חלון קבוע בעל הערך 7 וערכים משתנים של :



בדוגמא זו, התוצאה הטובה ביותר התקבלה עבור הפרמטרים ו- בה הרעשים הושתקו, הקצוות נשמרו, ולא נוצר טשטוש או מריחה חזקה.

בשורה השלישית, בה הגדלנו את ערך, התמונות יצאו מטושטשות. לכן נסיק שעבור , על פי התמונות שבשורה השנייה, נקבל את התוצאה האופטימלית.

עוד ניתן לראות כי עבור שינוי הערכים בעמודות ערכו של פחות משפיע על התוצאה הסופית.

נשים לי כי עבור השורה הראשונה בה נמוך, אנו ממצעים רק פיקסלים בצבעים דומים, ולכן אם במצב זה נגדיל את , התמונה לא תשתנה משמעותית.

בנוסף ניתן לראות כי בתמונות שבעמודה הראשונה, בה נמוך, גם כש- גבוה והתמונה מטושטשת, הקצוות נשמרים. לעומת זאת, בעמודה האחרונה, בה גבוה, משפיע מאוד. ההבדל נובע מכך שהפונקציה נותנת לכל הפיקסלים השכנים משקל דומה, ולכן הערך החדש עבור כל פיקסל מושפע מעיקר מהפונקציה .

נראה דוגמות נוספות לתמונות שהתקבלו על ידי הפעלת האלגוריתם, בה השתמשנו בערכים האופטימליים עבור הפרמטרים ל-Gaussian שמצאנו.

תמונה בגווני אפור:





התמונה לאחר הפעלת האלגוריתם תוך שימוש בפרמטרים ו-

התמונה המקורית

תמונה צבעונית:



התמונה המקורית

התמונה לאחר הפעלת האלגוריתם תוך שימוש בפרמטרים ו-

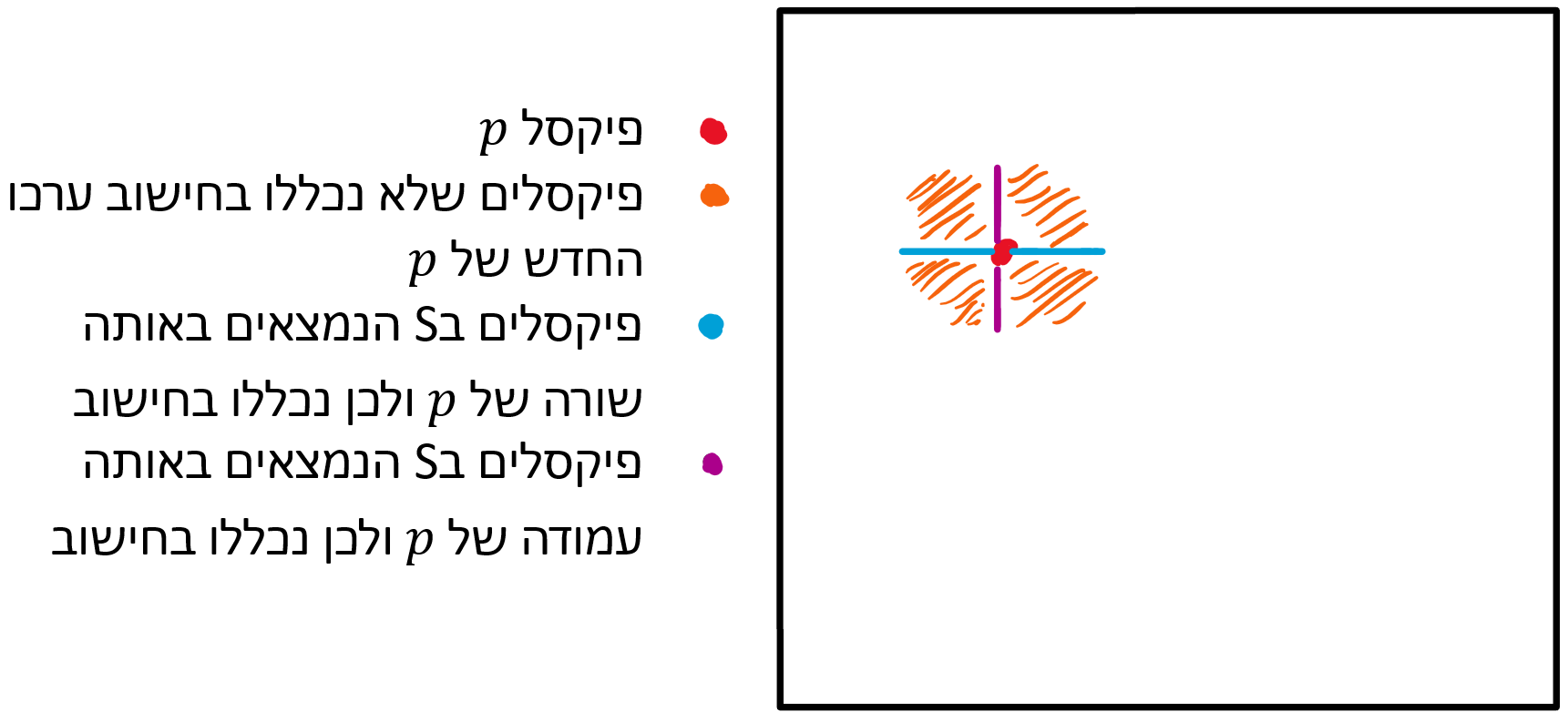
**רעיון לייעול**

באלגוריתם שתיארנו עד כה, ערכו של כל פיקסל הוא ממוצע משוקלל של ערכי כל הפיקסלים בחלון ריבועי סביב בתמונה. זמן הריצה של חישוב ערך כל פיקסל בתמונה הוא: ( כאשר S היא קבוצת הפיקסלים בחלון הריבועי.

קיימת אפשרות לייעול זמן הריצה, בה נכלול בחישוב הממוצע המשוקלל רק פיקסלים שנמצאים באותה שורה או באותה עמודה של הפיקסל . לפי שיטה זו נחשב תחילה את ערכו של כל פיקסל בתמונה לפי ממוצע משוקלל של הפיקסלים בתחום S הנמצאים באותה השורה של פיקסל p, ולאחר נחשב ממוצע משוקלל של ערכי הפיקסלים באותה העמודה של p בתחום S, תוך שימוש בערכים שהתקבלו בשלב הראשון.

בשיטה זו עבור כל פיקסל מתבצעים שני חישובים חד-מימדיים זולים יחסית במקום חישוב דו-מימדי אחד יקר.

זמן הריצה של חישוב ערך כל פיקסל בתמונה עם שינוי זה יהיה: כאשר S היא קבוצת הפיקסלים בחלון הריבועי.



מקורות:

* <https://people.csail.mit.edu/sparis/bf_course/> -A gentle introduction to BF
* [https://web.archive.org/web/20120511061328/http://scien.stanford.edu/pages/labsite/2006/psych221/projects/06/imagescaling/bilati.html](https://web.archive.org/web/20120511061328/http:/scien.stanford.edu/pages/labsite/2006/psych221/projects/06/imagescaling/bilati.html)
* Bilateral Filtering for Gray and Color Images by C. Tomasi and R. Manduchi
* <https://people.cs.clemson.edu/~dhouse/courses/404/lectures/bilateral/06_implementation.pdf> תחת Separable Kernel [Pham and Van Vliet 05]
* <https://www.researchgate.net/publication/4181202_Separable_bilateral_filtering_for_fast_video_preprocessing>