

## סיכום כללי ראייה ממוחשבת ועיבוד תמונה:

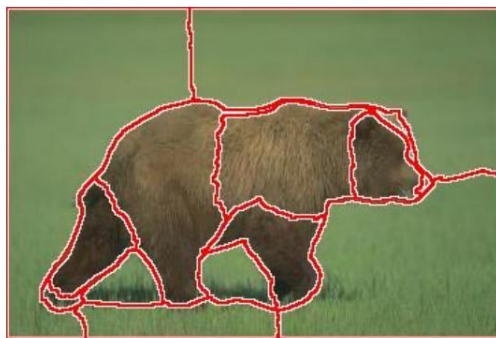
### נושאים (מצגת ראשונה):

1. תיקון גמא
2. איזון הסטנוגרמה
3. כימות

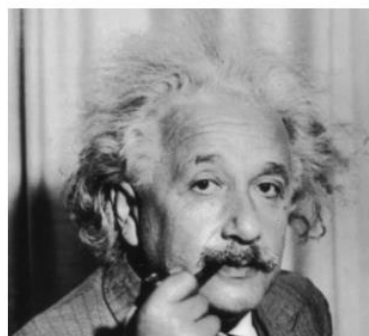
### הבדלים בין עיבוד תמונה לראייה ממוחשבת:

**ראייה ממוחשבת (computer vision)** – המטרה היא לזהות עצמים מתמונות / הבנה של "מה קורה בתמונה", בראייה ממוחשבת קיימים כמה סוגים של זיהוי:

1. חלוקה (segmentations) – לדעת לחלק את התמונה למספר קטעים, למשל להבדיל בין דשא לדוב או זברות כמו:



2. הכרה (recognition) – לזהות מה הוא האובייקט הנמצא בתמונה, למשל, האם איינשטיין נמצא בתמונה הבא?



### אתגרים בזיהוי / הבנה של תמונות:

1. נקודות מבט
2. תאורה
3. קנה מידה
4. דפורמציות (עיוות צורות)
5. אובייקטים החוסמים אובייקטים אחרים
6. רקע
7. תמונה

**עיבוד תמונה (image processing)** – לקחת תמונה ולשנות את הדרך שהיא נראת, בדרך כלל משינוי ישיר של הפיקסלים המרכיבים את התמונה, תמונה בסופו של דבר היא מטריצה שכל תא בה מייצג פיקסל בתמונה, עוד מטרה מרכזית בעיבוד תמונה היא להפוך תמונה ל-"טובה יותר" כך שיראו אותה יותר טוב בין אם זה הבהרה

### כימות:

מה המספר המקסימלי של ערכים שיכול להיות בפיקסל?

תמונה עם 1 ביטים  $2^1 = 2 \rightarrow 0 - 1$

תמונה עם 8 ביטים  $2^8 = 256 \rightarrow 0 - 255$

תמונה עם 16 ביטים  $2^{16} = 65536 \rightarrow 0 - 65536$

כשמדברים על כימות שואלים מה מספר האפשרויות לערכים עבור פיקסל  $\langle x, y \rangle$ , למשל הנה תמונה כך שכל פיקסל בה הוא ערך יחיד המביע "כמה לבן נשים בפיקסל" כאשר בתמונה מימין יש רק 2 אפשרויות לערכים, משמאלה 4, לאחר מכן 8 ואז 256, כלומר בתמונה הראשונה כל פיקסל יכול ערך שהוא 0 (שחור) או 1 (לבן), ובתמונה האחרונה אנחנו מדברים על סקאלה של בין 0 ל-255, כאשר 0 הוא הכי שחור ו-255 הוא הכי לבן:



256 Levels

8 Levels

4 Levels

2 Levels

כאשר לתמונות יש צבעים אז הפורמט הכי מוסכם ונפוץ במחשב הוא RGB (red, blue, green), כלומר לכל פיקסל יש 3 שכבות שכל שכבה היא אחת מהצבעים הנ"ל ויש לה ערך כמה מהצבע לשים בפיקסל.

בתמונה שהיא שחור לבן יהיה לנו מטריצה בגודל  $M \times N$ ,

ובתמונה שהיא עם צבעים יהיה לנו מטריצה בגודל  $M \times N \times 3$ .

### פעולות מבוססות נקודות (point-based operations):

פעולה התלויה בערך של הפיקסלים, כל פעולה על פיקסל אינה תלויה בערכים של הפיקסלים האחרים בתמונה או במיקום שלהם:

$$g(x, y) = \alpha \cdot f(x, y) + \beta$$

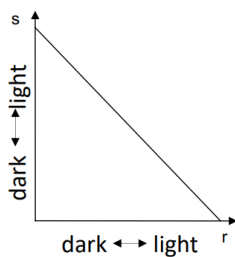
## טרנספורמציה בעוצמה (Intensity Transformation):

טרנספורמציה בעוצמה היא פונקציה שממפה בין קלט יחיד לפלט יחיד, סימון:  $s = T(r)$ ,

### דוגמאות:

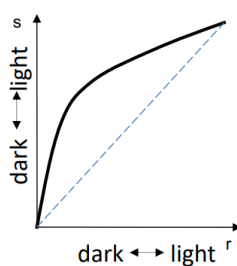
1. ניגודיות של תמונה:

$$s = 1 - r$$



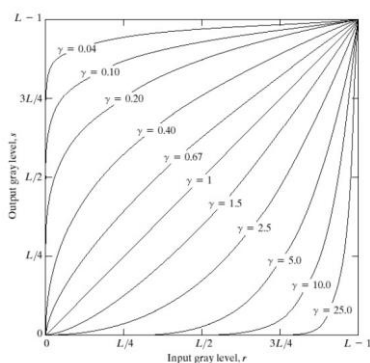
2. טרנספורמציה לוגריתמית:

$$s = c^{\log(1+r)}$$



### 3. טרנספורמציה חזקה:

$$s = c \cdot r^\gamma$$



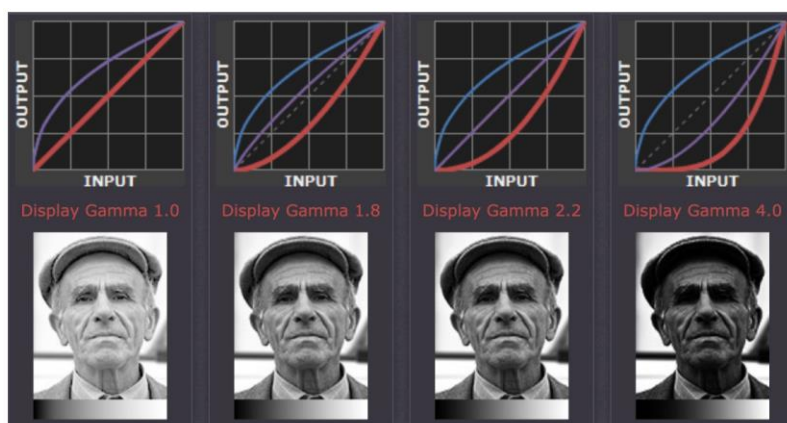
$\gamma = 3, 4, 5$



### קידוד ותיקון (Encoding & Correction):

נוכל להשתמש בעזרת טרנספורמציה בעוצמה בפיקסלים כהים ספציפיים בתמונה עם יותר ביטים ובחלקים בהירים ספציפיים בתמונה עם פחות ביטים,  $\frac{1}{2.2} \sim 0.45$ , כתוצאה נוכל לשמור יותר את הפרטים הכהים בתמונה, המטרה היא לתאם בין התמונה לבין הרגישות שבה רואה העין האנושית.

ברגע שנשמור את התמונה במחשב היא לא תשמור את השינויים שביצענו בה כי זה רק קידוד ולכן בצורה דפולטיבית המסכים בדרך כלל יתקנו את התמונה ויקודדו אותה עם גמא של 2.2, דבר זה נקרא sRGB standard.



## שוויון הסטוגרמה (Histogram Equalization):

המטרה היא להפוך תמונות מעורפלות לתמונות שנראות יותר טוב לעין, למשל:



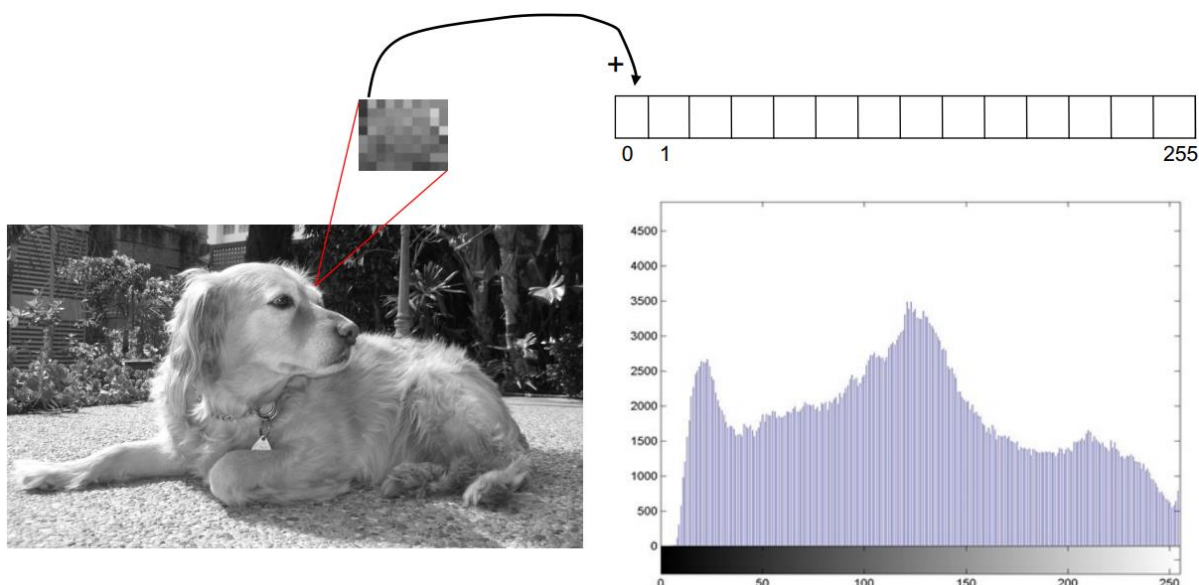
נרצה להגביר את הניגודים שבתמונה.

ניגודים של תמונה הם:

- היחס בין הבהירות לכהות.
- ההפרש בין המקסימום והמינימום של העוצמה של תמונה.
- יש קשר להסטוגרמה של התמונה.

## הסטוגרמה של תמונה (Image Histogram):

הסטוגרמה היא קירוב משוער של ערכים המתפלגים בצורה נורמלית ומיוצג בצורה של גרף, הסטוגרמה של תמונה היא תצוגה גרפית של התפלגות של צללים וערכים של פיקסלים של תמונה, לפי טווח הצבעים הנמצאים בתמונה, למשל:

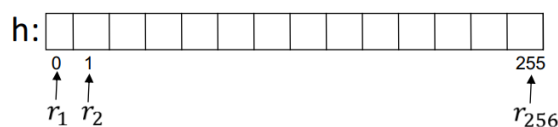


בגלל שהפיקסלים של התמונה הזאת הם בטווח של 0 עד 255 (כי היא בצבע שחור לבן) אז ניתן להציג את ההסטוגרמה של התמונה כך שנציג כל פיקסל מה הערך שלו בין 0 עד 255 וכך נראה איך הערכים של הפיקסלים מתפלגים בתמונה.



## הסטנוגרמה להסתברות:

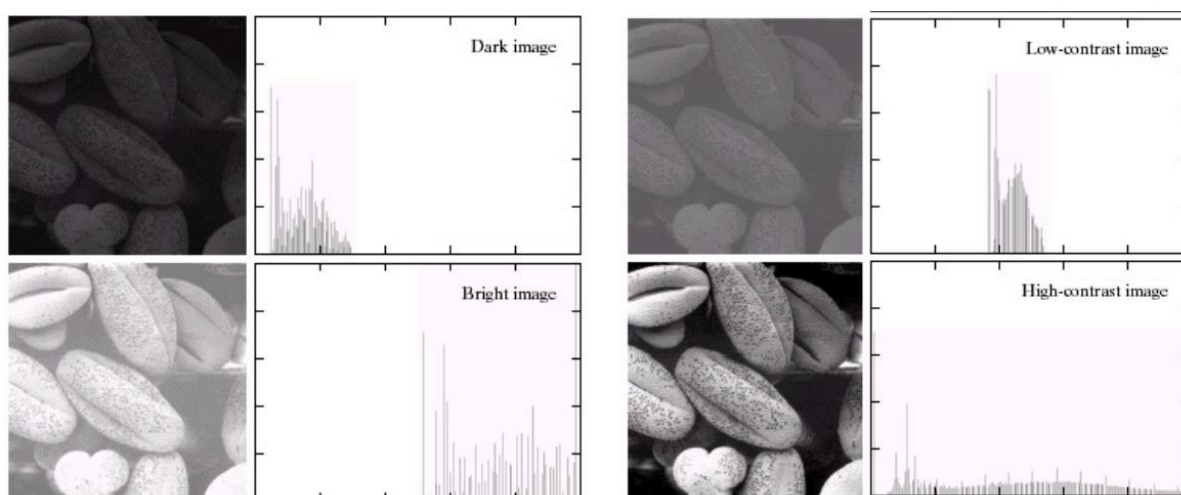
לכל עמודה בגרף כלומר לכל ערך בטווח של התמונה (למשל בין 0 עד 255) נגדיר  $r_k$  כאשר  $k$  הוא מספר העמודה בגרף כך שבמקרה שלנו  $1 \leq k \leq 256$ ,



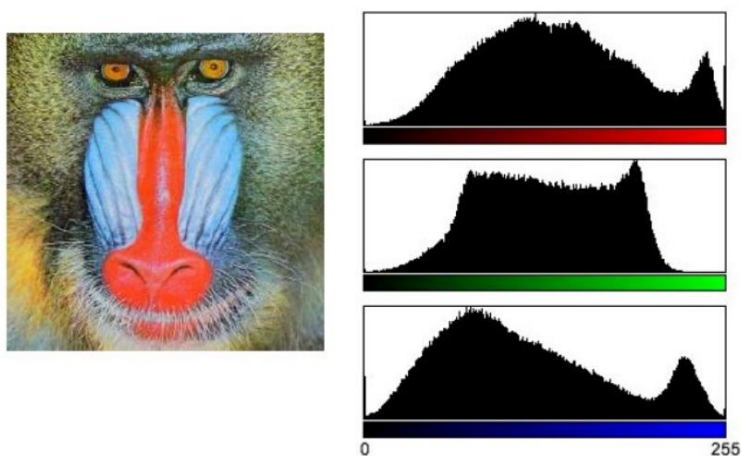
הפונקציה  $h$  הוא פונקציית הסטנוגרמה הנותנת כמה פיקסלים מתפלגים יש באותה עמודה  $h(r_k) = n_k$ , כעת נרצה להגדיר פונקציית הסתברות  $p$  הנותנת מה ההסתברות לאותו ערך בפיקסל, ולכן נגדיר את  $p$  לפי:

$$p(r_k) = \frac{n_k}{N}$$

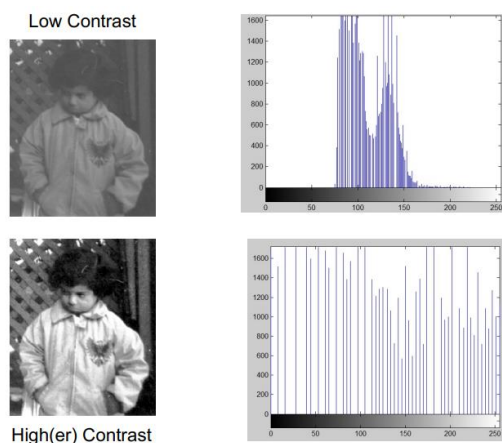
לפי ההסטנוגרמה נוכל להבין איך התמונה נראת פחות או יותר מבחינה התפלגות של הערכים של הפיקסלים של התמונה:



הסטנוגרמה של תמונה צבעונית מתחלקת ל-3 גרפים שונים, אחת לכל צבע:



המטרה שלנו היא לשפר ההסטנוגרמה של התמונה, כלומר לגרום לערכים של הפיקסלים להתפלג בצורה יותר טובה, למשל:



נרצה להפוך את התמונה העליונה לכמו התחתונה.

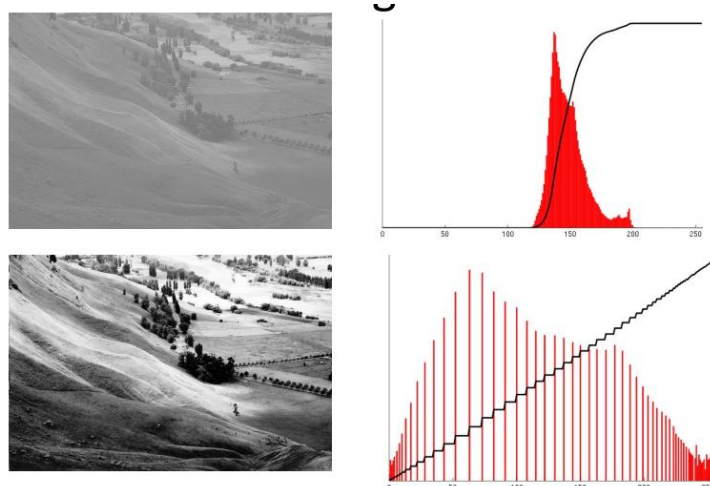
איך נשפר את ההסטנוגרמה?

1. נבין את הפרמטרים של התמונה.
2. ניצור מודל (עם נוסחה) המתאר את בעיה בשימוש במאפיינים/פרמטרים/תכונות מסוימות.
3. נגדיר פתרון המבוסס על נוסחת המודל.
4. ניצור אלגוריתם לפתרון (מזעור) של נוסחה.

### מודל ניגודיות (Contrast Model):

נמצא מודל המדמה את המציאות שלנו לפי איך שאנחנו רואים אותה, נרצה לסווג את ההסטנוגרמה כ-"טובה" או "רעה" לפי זה.

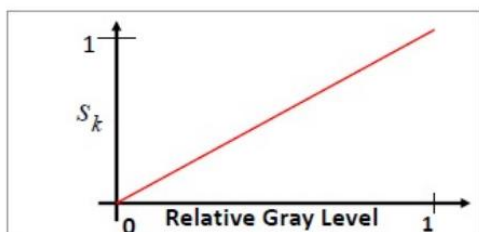
קודם נסתכל על הקשר בין הסטנוגרמה לבין ניגודיות:



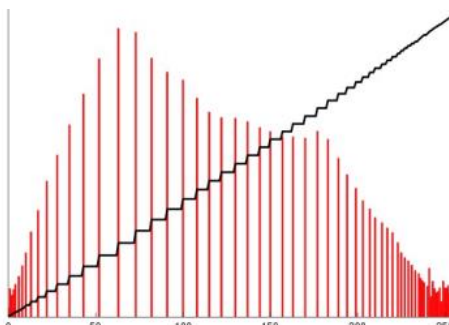
### נסתכל על התפלגות של הסטנוגרמה מנורמלת (CDF):

נניח וקיימים  $N$  פיקסלים, התמונה היא בשחור לבן (greyscale) בין ערכים 0 עד  $k$ ,  $n_i$  הוא ערך הפיקסלים במקום ה- $i$  בהסטנוגרמה, ננרמל לפי:

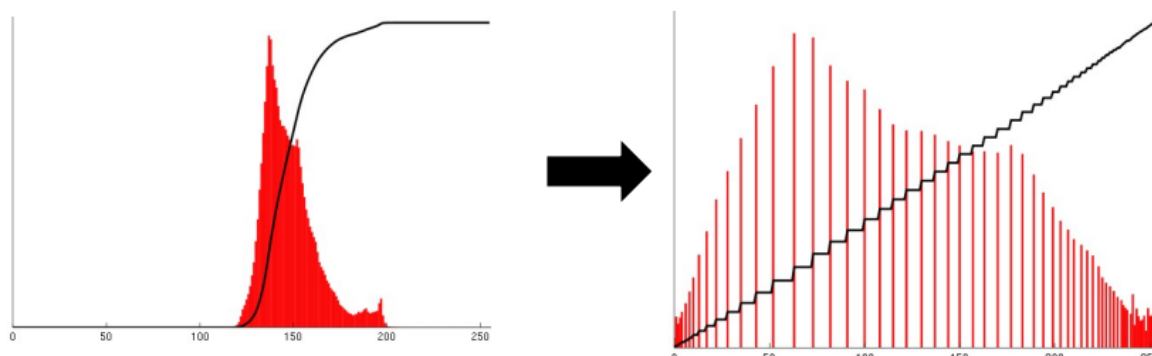
$$S_k = \frac{1}{N} \sum_{i=1}^k n_i$$



לפי הנרמול ניתן להגיד שניגודיות טוב היא הסטנוגרמה ליניארית:



המטרה של שוויון הסטנוגרמה:



### שוויון הסטנוגרמה:

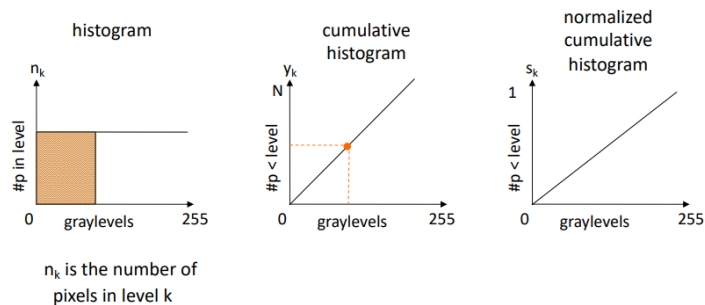
- המטרה: לשפר ניגודיות.
- מודל: ניגודיות של הסטנוגרמה של תמונה (CDF).
- בעיה: ערך של פיקסלים (במקרה שלנו *grey level*) מפוזרים בצורה לא שווה בתמונה.
- פתרון: שימוש שווה בין כל הערכים בתמונה.
- איך? להפוך את ההסטנוגרמה שתתפלג בצורה נורמלית ובצורה ליניארית.



## מודל רכיבים (Model Components):

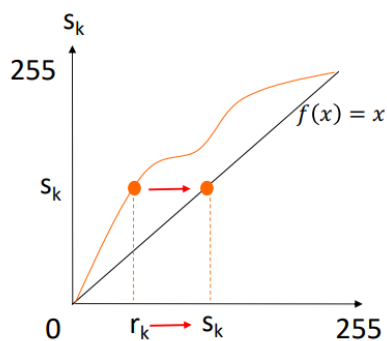
נניח ויש לנו תמונה עם  $N$  פיקסלים וערכים בין 0 ל- $k-1$  (נניח במקרה שלנו  $k = 255$ ), שלושת הרכיבים שלנו הם:

1. הסטנוגרמה.
2. הסטנוגרמה צפיפות.
3. הסטנוגרמה צפיפות מנורמלת.

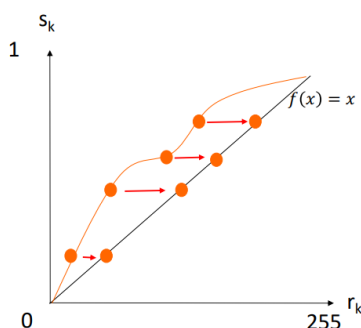


## פתרון של שוויון הסטנוגרמה:

נגרום לשינוי בעצמה של התמונה כך שההתפלגות של הגרף ההסטנוגרמה בתמונה יהיה ככל האפשר לצורה ליניארית, יותר נכון להגיד שיהיה מקורב לפונקציית הזהות  $f(x) = x$ , נמפה את כל הרמות  $r_k$  ל- $s_k$ , ונגרום להתפלגות של ההסטנוגרמה להיות כמו הגרף:

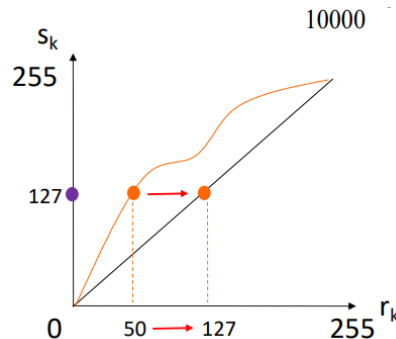


נעשה זאת לכל רמות הערכים של הפיקסלים בהסטנוגרמה:



איך נבצע את הנרמול? ניקח את הערך של אותו  $r_k$ , נחלק אותו במספר הפיקסלים כפול הערך המקסימלי של השלבים של התמונה (למשל 255), לדוגמה, נניח ויש לנו  $N = 10,000$  פיקסלים בתמונה, הערך המקסימלי של  $k$  הוא 255, נניח והערך הנוכחי  $r_k = 5000$  אז ננרמל את אותו פיקסל לפי:

$$\frac{r_k}{N} \cdot \max(k) = \frac{5000}{10000} \cdot 255 = 127$$



### בצורה פורמלית:

לכל פיקסל שאנחנו מנרמלים את ההסטוגרמה שלו נחשב לפי:

$$cdf(R) = \sum_g^R p(g)$$

כאשר יש  $G$  שלבים של אפור,  $R$  ערך קודם של השלב של הפיקסל,  $S$  השלב החדש של הפיקסל, אז קיימים לנו את התנאים הבאים:

1. אנחנו נשנה את  $R$  ל- $S$  כך:

$$\sum_{t=0}^S P_{new}(t) = \sum_{g=g_{min}}^R P_{old}(g)$$

2. העצמה ההסתברותית החדשה צריכה להיות שווה:

$$P_{new}(t) = \frac{1}{G}$$

הפתרון (כאשר  $R = g_{old}, S = g_{new}$ ):

$$g_{new} = \left\lceil G \sum_{g=g_{min}}^{g_{old}} p_{old}(g) - 1 \right\rceil$$

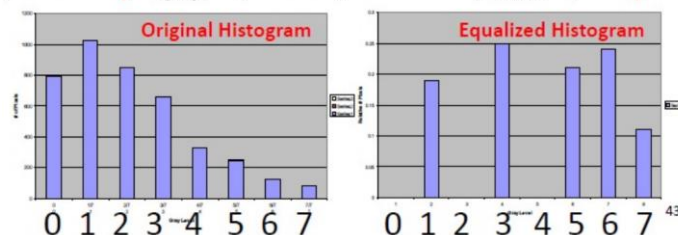
נשיב לב:

$g_{new}$  הוא לא ידוע,  $g_{old}$  הוא ידוע

$p_{old}(g)$  הוא ההסטוגרמה של הפיקסל בשלב  $g$

דוגמה:

Grey Level ( $k$ )	Normalized ( $r$ )	# Pixels ( $n$ )	Normalized ( $n/N$ )	Cumulative	Approx	Result
0	0	790	0.19	0.19	1/7	1
1	1/7	1023	0.25	0.44	3/7	3
2	2/7	850	0.21	0.65	5/7	5
3	3/7	656	0.16	0.81	6/7	6
4	4/7	329	0.08	0.89	6/7	6
5	5/7	245	0.06	0.95	1	7
6	6/7	122	0.03	0.98	1	7
7	7/7	81	0.02	1.00	1	7
Total:		4096	1			



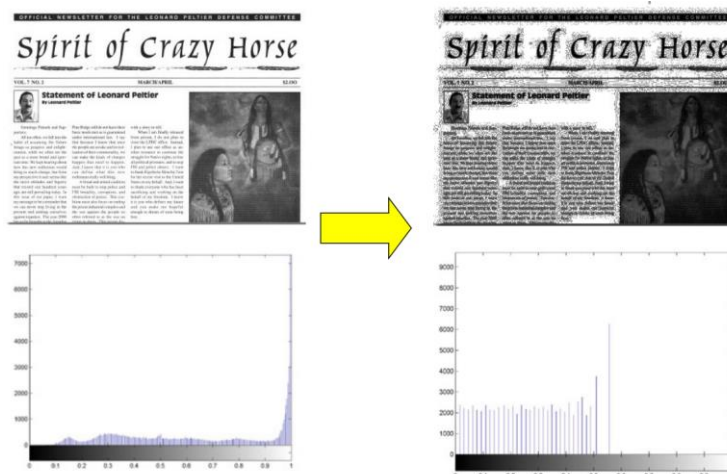
### אלגוריתם עבור שוויון הסטנוגרמה (Histogram Equalization):

1. חשב את ההסטנוגרמה של התמונה.
2. חשב את ההסטנוגרמה המצטברת של התמונה (cumulative):
3. נרמל את ההסטנוגרמה המצטברת ע"י חלוקה במספר המקסימלי של הפיקסלים.
4. הכפל את הערכים של ההסטנוגרמה המנורמלת בערך המקסימלי של השלבים  $(k - 1)$ .
5. עגל את המספרים על מנת לקבל מספרים שלמים.
6. מפה את העצמה של הערכים של התמונה בשימוש התוצאות משורה 5.
7. וודא שהערך המינימלי הוא 0 והערך המקסימלי הוא  $k - 1$ , אחרת פזר את התוצאות בצורה ליניארית על הטווח  $[0, k - 1]$ .

מתי האלגוריתם ייכשל?

- כאשר השלב של הערכים של הצבעים הרצוי לא מתפלג בצורה נורמלית, למשל טקסט.
- כאשר שני תכנים שונים משולבים ביחד (2 תמונות או תמונה וטקסט וכו').

למשל:



**אלגוריתם משופר עבור שוויון הסטנוגרמה (Adaptive Histogram Equalization):**