אלגוריתם Neural Style Transfer

מבוסס על המאמר: <https://arxiv.org/pdf/1508.06576.pdf>

רשת הנוירונים היעילה ביותר לצרכי עיבוד תמונה\ראייה מממוחשבת היא רשת קונבולוציה (CNN). רשת קונבולוציה בנויה משכבות קטנות שמעבדות את המידע הויזואלי בתמונה. כל שכבה מכילה אוסף של פילטרים, שכל אחד מהם לומד feature ספציפי מתמונה הכניסה. הפלט של כל שכבה נקרא feature map.

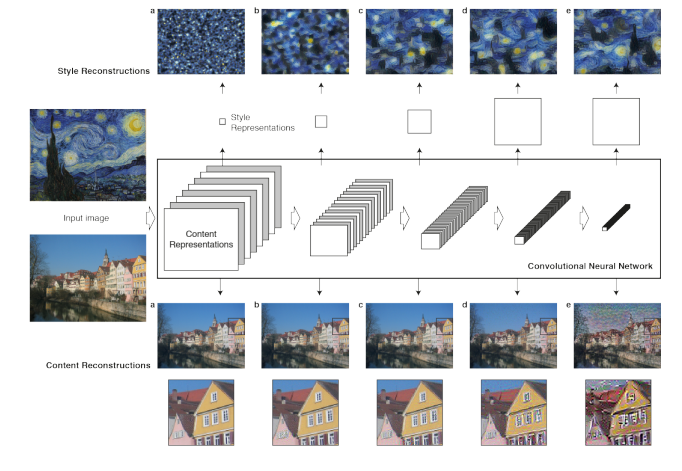
כאשר רשת קונבולוציה מאומנת לזיהוי אובייקט בתמונה, השכבות העליונות של הרשת יילכדו תוכן שהוא High Level במונחים של האובייקט, והשכבות הנמוכות יילכדו תוכן שהוא Low Level, ברמת הפיקסל. לכן, נתייחס לתגובות ה – features בשכבות העליונות של הרשת כ – content representation.

כדי לקבל style representation, נשתמש ב – feature space שתוכנן ללכוד מידע על הטקסטורות. מרחב ה – features הזה נבנה על תגובות הפילטרים של כל שכבה ברשת, ומורכב מהקורלציה בין תגובות הפילטרים השונות. באופן זה נקבל ייצוג multi scale סטציונרי של תמונת הכניסה שלוכדת את המידע על הטקסטורות, אבל לא את הסידור הגלובלי.

תמונת כניסה ניתנת לייצוג ע"י קבוצת פילטרים בכל שכבה ב – CNN. בכל שלב מספר הפילטרים גדל, אבל גדלים קטן כתוצאה מפעולות שונות שמבצעות downsampling – כמו pooling.

Content Reconstruction – ניתן לבצע ויזואליזציה של המידע בשלבים שונים ב – CNN ע"י בניית תמונת הכניסה מתגובות הפילטרים בשכבה מסויימת. הייצוג בשכבות הנמוכות הוא כמעט מושלם, כי הוא ברמת הפיקסל, בעוד שבשכבות הגבוהות יש איבוד מידע, ונשמר רק מידע שהוא High Level.

Style Reconstruction – מעל הייצוגים המקוריים של ה – CNN, נבנה Feature Space חדש שלוכד את ה – Style של תמונת כניסה. ה – Style Representation מחשב את הקורלציה בין features שונים בשכבות שונות ב – CNN.



באמצעות ה – Style Representation וה – Content Representation, ניתן לבנות תמונות חדשות שמהוות ערבוב של שני הייצוגים האלה. התמונות החדשות נוצרות ע"י מציאת תמונה שמתאימה באופן אופטימלי לשני הייצוגים. המבנה הגלובלי של התמונה נשמר, אך הצבעים והטקסטורות הם אלו שמשתנים.

כאשר מבצעים התאמה בין ה – Style Representations בשכבות הגבוהות של הרשת, מבנים מקומיים בתמונה מותאמים ויוצרים תמונה חלקה ורציפה יותר.

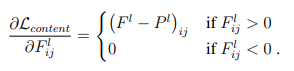
השיטה:

רשת הקונבולוציה שמשתמשים בה לאלגוריתם היא VGG-16 – רשת קונבולוציה שמשתמשים בה לצרכי זיהוי אובייקטים. נשתמש ב – feature space של 16 שכבות קונבולוציה ו – 5 שכבות pooling. אם נחליף את ה – max pooling ב – average pooling נקבל תוצאות טובות יותר. כל שכבה ברשת מגדירה filter bank לא לינארי. לכן, תמונת כניסה מקודדת בהתאם לפילטרים בכל שכבה. שכבה שמכילה פילטרים שונים זה מזה מכילה feature maps שונים בגודל כל אחד (ע"י חישוב של גובה x רוחב של ה – feature map). לכן, ניתן לשמור תגובה של שכבה l במטריצה , כאשר זאת האקטיבציה של הפילטר ה - i במיקום j בשכבה l. כדי לבצע ויזואליזציה של התמונה שמקודדת בשכבות שונות, נבצע gradient descent על תמונת רעש לבן כדי למצוא תמונה אחרת שמתאימה ל – feature responses של התמונה המקורית. תהי התמונה המקורית ו - התמונה החדשה שנוצרה. ו - אלו ה – feature representation בשכבה ה – l.

נגדיר את ה – squared error loss בין ה – feature representations:



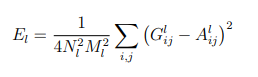
נגזור:



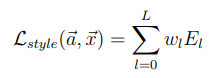
עבור ה – Style – מעל תגובת ה – CNN בכל שכבה נבנה Style Representation שמחשב את הקורלציה בין תגובות פילטרים שונות, כאשר התוחלת מחושבת באופן מרחבי על תמונת הכניסה. הקורלציה בין ה – features ניתנת להצגה ע"י Gram Matrix – , כאשר הוא:



כדי לייצר טקסטורה שמתאימה לסטייל של תמונה מסויימת, נשתמש באלגוריתם הגרדיאנט, כאשר הכניסה תהיה רעש לבן, והמטרה תהיה למצוא תמונה אחרת שמתאימה ל – Style Representation של התמונה המקורית. זה נעשה ע"י מזעור ה – mean squared distance בין ה – Gram Matrix של התמונה המקורית לבין ה – Gram Matrix של התמונה החדשה. תהי התמונה המקורית ו - התמונה החדשה שנוצרה. ו - אלו הן ה – Gram Matrix בשכבה ה – l. תרומה של כל שכבה ל – loss הכולל היא:

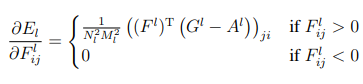


ה – loss הכולל הוא:

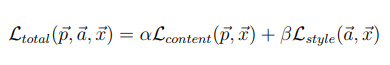


כאשר ממשקלים את השכבות ע"י .

נגזור ונקבל:

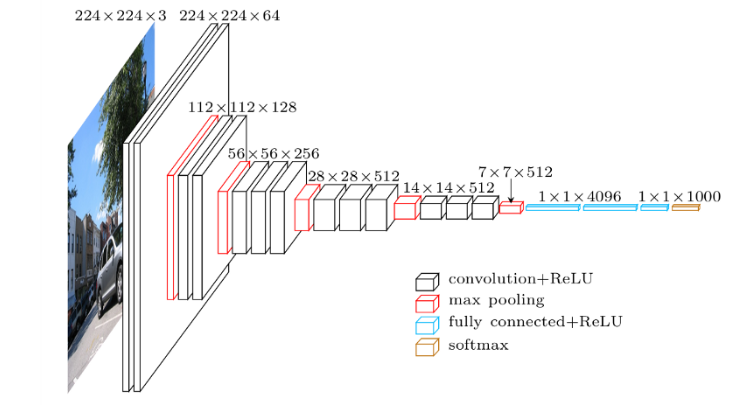


כדי ליצור תמונה שמכילה ערבוב בין תמונת תוכן לתמונת סטייל נרצה למזער את המרחק בין תמונת רעש לבן מה – Content Representation של התמונה בשכבה אחת ואת ה – Style Representation במספר שכבות. תהי תמונת תוכן, ו - התמונה הסופית שנרצה לקבל. נרצה למזער את פונקציית המחיר:



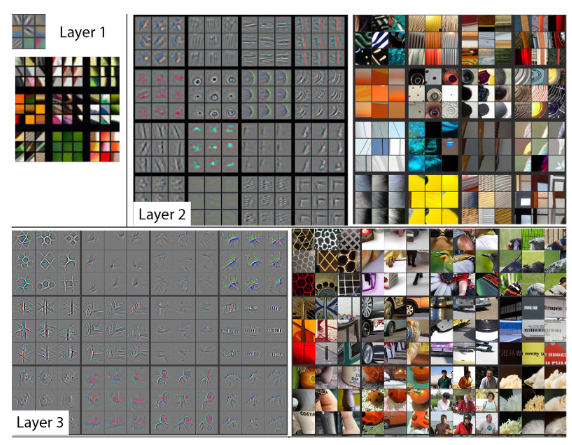
*כאשר α, β ממשקלים את תמונת התוכן ואת תמונת הסטייל בהתאם.*

*תיאור האלגוריתם עם רשת* VGG-16*:*



*הקונבולוציה הראשונה משתמשת בפילטרים בגודל* 3x3 – *64 פילטרים כאלה ליצירת 64* feature maps.

*ה –* feature maps *ייראו כך:*



קישורים רלוונטיים:

* <https://towardsdatascience.com/neural-style-transfer-tutorial-part-1-f5cd3315fa7f>
* <https://ayearofai.com/rohan-lenny-2-convolutional-neural-networks-5f4cd480a60b>
* מימוש Artistic Neural Style Transfer ב – PyTorch:

<https://www.kaggle.com/soumya044/artistic-neural-style-transfer-using-pytorch/notebook>