אופיר בירקה - 316389410
נריה אוחנה - 304933575

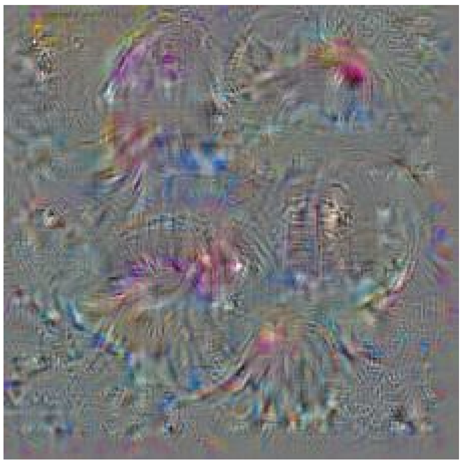
<u>מבוא לרשתות נוירונים תרגיל 2</u>

<u>חלק מעשי:</u>

שאלה 1:

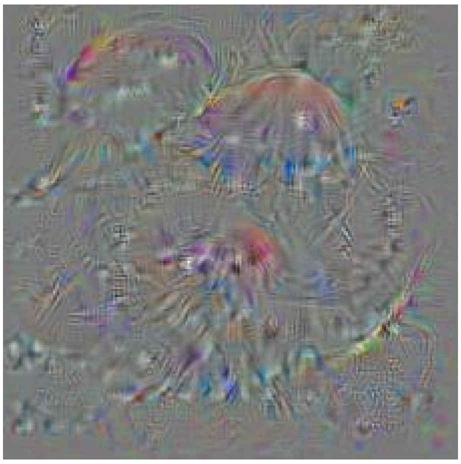
על מנת לאמן קלט שיתן ערך מקסימלי לנוירון מסוים, הגדרנו את הקלט של הרשת כמשתנה מסוג tf.variable, ואת כל המשקולות ברשת הגדרנו כקבועים, כך שלא ישתנו במהלך האימון.
כדי להציג נוירון משכבת קונבולוציה, בחרנו בנוירון שנמצא בשכבה השניה במיקום [0,0,0,0] והגדרנו רגולריזציה כנורמה בריבוע של תמונת הקלט, בעקבות המאמר של Simonyan et al. 2014.
את הרשת אימנו בעזרת GradientDescentOptimizer עם learning rate של 0.33
פונקציית האופטימיזציה שמצאנו על ידי ניסוי וטעיה , היתה
min(- neuron_val *1000 + norm^2)
ואחרי 1000 איטרציות קיבלנו:

ניתן לראות שהרשת עשתה שינויים בחלק השמאלי העליון, שהוא משפיע על נוירון בשכבת הקונבולוציה השניה במיקום [0,0,0,0] , ויצרה ריבועים קטנים בתוך חלק זה. אך בשאר התמונה שמה אפסים כדי שהרגולריזציה תשאר נמוכה.
כדי להציג ויזואלית נוירון משכבת FC , בחרנו בשכבה האחרונה של הרשת (FC8), ובחרנו בנוירון שמזהה מדוזות (107). ניתן לראות שכאן כל שטח התמונה הושפע מהאימון, ויצר צורות שונות של מדוזות על פני שטח התמונה:



<u>שאלה 2</u>

כדי שהתמונה תראה דומה לתמונה טבעית, יצרנו תמונה שעבוד כל תדר של טרנספורם פוריה, מכילה את הערך הממוצע לתדר בתמונות טבעיות, ובמהלך האימון של הרשת הוספנו רגולריזציה שמשווה את הטרנספורם פוריה של התמונה לערך הממוצע, כך שהנורמה של ההפרש תהיה קטנה ככל האפשר.
בצענו את הניסוי על אותו נוירון ב FC8 שמייצג מדוזה, וקיבלנו תמונה קצת יותר חלקה ודומה לתמונה טבעית:



<u>שאלה 3</u>

גם פה הגדרנו את הקלט כ tf.variable ושמרנו על המשקולות כקבועים.
אתחלנו את הקלט עם תמונה של מדוזה, וניסינו לאמן את הרשת כך שתיצור תמונה דומה למדוזה (על ידי רגולריזציה של נורמה של ההפרש בין התמונות) אך שהרשת תזהה אותה כמו Mink (סוג של מכרסם).
הפונקציה שאימנו:

min(- mink_prob * 500000 + norm (image - original_image))

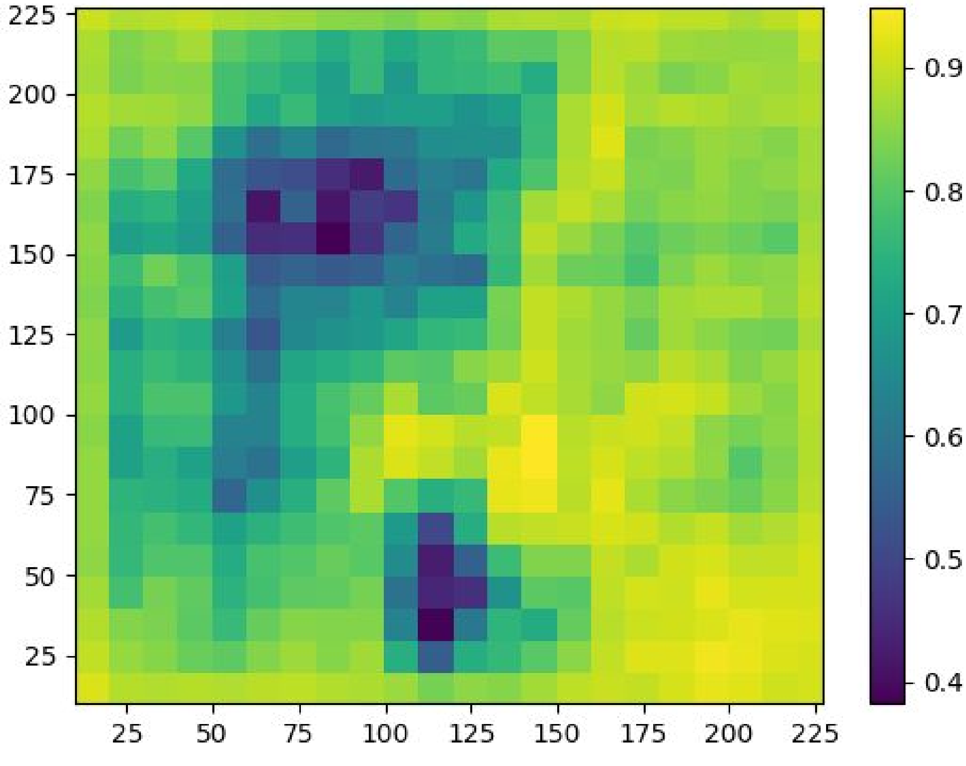
ואכן קיבלנו תמונה שנראית ויזואלית דומה למדוזה, אך ה AlexNet מזהה כמדוזה:

התמונה המקורית זוהתה כמדוזה בהסתברות של : 0.583915
התמונה החדשה שקיבלנו זוהתה כ Mink בהסתברות: 0.977706

שאלה 4

המשכנו עם תמונה של מדוזה, ובדקנו מה קורה כשאנו מסתירים בלוקים שונים מתוך התמונה. עבור כל בלוק בדקנו מה אחוז הזיהוי של התמונה כאשר מאפסים אותו , וייצגנו את ההסתברות שקיבלנו בצבע מתאים כדי לקבל מפת חום. עבור התמונה:

קיבלנו מפת חום:



האזור המרכזי שבו הזיהוי נמוך נמצא בחלק העליון כלפי שמאל, שזה מקביל לאזור שבו נמצא גוף המדוזה בתמונה, ולכן אם מאפסים בלוק באזור זה הזיהוי ירד משמעותית, לעומת אזורים אחרים שבהם זה פחות משנה.

<u>חלק תיאורטי</u>

שאלה 1:

ה receptive field בכל רשת יהיה:

1. 7x7=49
2. (3+2+2)x(3+2+2) = 49
3. (5+2)x(5+2) = 49
4. (3+2+2+2)x(3+2+2+2) = 9x9 = 81

כמה משקולות וכמה מכפלות בתמונה שגודלה nxn ברשתות 1,3:

- ברשת (1): נצטרך 49 משקולות, ונצטרך 7x7x(n-6)x(n-6) מכפלות
- ברשת (3): נצטרך 9+25=34 משקולות ונצטרך 3x3x(n-6)x(n-6)x5x5x(n-4)x(n-4)

ניתן לראות שככל שהstride גדל הreceptive field של כל נוירון גדל כיוון שכאשר עושים קונבולציה למשל עם פילטר 3x3 עם stride=1 יש 2 שורות/עמודות חפיפה בין הreceptive field של כל נוירון. אם היינו לוקחים stride יותר גדול, בלי אזורי חפיפה אז הreceptive field של כל נוירון ונוירון היו שונים, ולכן האזור שמשפיע על הנוירון בתמונה המקורית גדול יותר.

שאלה 2

נאתר מבנה דומה לרשת העמוקה שראינו בכיתה:

נניח $f(x) = \sum_{i=1}^{n} a_i\,\sigma(w_i x + b_i)$

הרשת העמוקה - בכל שכבה יהיו 3 נוירונים h_1, h_2, h_3. נתאר שיטה איטרטיבית לבניית הרשת העמוקה:

בשכבה הראשונה:

1. אם $a_1 > 0$ אז $h_1 = a_1\sigma(w_1 x + b_1)$ אחרת $h_3 = -a_1\sigma(w_1 x + b_i)$
2. נעביר את x

בשכבה i:

1. אם $a_i > 0$ אז $h_1 = h_1 + a_i\sigma(w_i x + b_i)$ אחרת $h_3 = h_3 - a_i\sigma(w_i x + b_i)$
2. נעביר את x

בסוף (השכבה האחרונה):

נחזיר את $h_1 - h_3$.

נשים לב שנקבל רשת עמוקה עם 3n נוירונים.

שאלה 3

גם אם האלגוריתם gradient descent יעצור אחרי מספר צעדים קבוע, נניח K, זה לא אומר שעל כל קלט נגיע לפתרון הבעיה תוך מספר צעדים זה. יתכן שהמרחק בין המשקולות ההתחלתיים לפתרון הנכון גדול יותר מגודל K צעדים. וכן יתכן שנעצור במינימום מקומי.

לכן ההנחה לא מוכיחה שקיים פתרון מהיר יותר לבעיה, וההוכחה מהכיתה על כך שבעיית האימון NP שלימה נשארת תקיפה.