סגמנטציה של מכשירים כירורגיים במהלך ניתוח באמצעות רובוט



מנחה: ד"ר הרמן מעינה

מגיש: טל רווה

**תוכן עניינים**

[**הקדמה** 3](#_Toc54360488)

[**סגמנטציה של מכשירים כירורגיים** 4](#_Toc54360489)

[**הגדרת הבעיה** 4](#_Toc54360490)

[**מוטיבציה** 5](#_Toc54360491)

[**הנתונים** 6](#_Toc54360492)

[**אתגרים** 8](#_Toc54360493)

[**ארכיטקטורות מבוססות FCN** 10](#_Toc54360494)

[**רשתות CNN** 10](#_Toc54360495)

[**רשתות FCN** 12](#_Toc54360496)

[**הארכיטקטורות השונות בתחרות EndoVis 2017** 15](#_Toc54360497)

[**ארכיטקטורת TernausNet-16** 24](#_Toc54360498)

[**ארכיטקטורת LinkNet-34** 27](#_Toc54360499)

[**תוצאות** 30](#_Toc54360500)

[**מדדי הערכת ביצועים** 30](#_Toc54360501)

[**ביצועי TernausNet-16 לעומת LinkNet-34** 31](#_Toc54360502)

[**תחרות EndoVis 2017** 33](#_Toc54360503)

[**סיכום, דיון, מסקנות** 38](#_Toc54360504)

[**מקורות** 40](#_Toc54360505)

[**מאמרים** 40](#_Toc54360506)

[**מקורות נוספים** 42](#_Toc54360507)

# **הקדמה**

ניתוחים המתבצעים באמצעות מכשירים כירורגיים המיועדים לניתוחים זעיר פולשניים קיימים מזה עשרות שנים. אחד היתרונות הגדולים של ניתוחים מסוג זה הוא האפשרות לבצע את הניתוח באמצעות מצלמה וכלים המוכנסים ישירות לגופו של המטופל, דרך נקבים קטנים יחסית, ללא צורך בפתיחה מלאה של חלק הגוף הרלוונטי – יכולת זו מאפשרת הקטנה של מידת הפגיעה ברקמות הגוף וקיצור זמן החלמה ואשפוז לאחר ניתוח. עקב כך, שיטות אלו הפכו למועדפות עבור ניתוחים רבים. עם זאת, ניתוחים אלו די מורכבים וכרוכים בהם קשיים ייחודיים. למשל, נדרשת מהמנתח רמת קואורדינציה גבוהה כדי לשלוט באופן טוב בכלים תוך כדי התבוננות בנראה במצלמה (בניגוד להתבוננות ישירה באברי הגוף). ניתן להשתמש בתמונות המתקבלות משימוש במצלמה כדי להתמודד טוב יותר עם קשיים מסוג זה ע"י הזנתן לאלגוריתמי למידת מכונה. אחת הבעיות החשובות שיש להתמודד איתן בהקשר הזה היא סגמנטציה של תמונה (Image segmentation) – שיטות הפותרות בעיה זו מהוות את אחד משלבי הביניים הדרושים עבור מערכות ממוחשבות אשר מטרתן לעזור למנתח. למשל, סגמנטציה יכולה להוות שלב מקדים לביצוע מעקב והערכת מיקום ומצב של הכלים הכירורגיים באיבר המטרה כדי למנוע טעויות חמורות כגון פגיעה באברים סמוכים.

סגמנטציה של תמונה היא בעיה ותיקה וקשה בראייה ממוחשבת אשר פותחו עבורה אלגוריתמים רבים במשך השנים. במקרה של סגמנטציה של מכשירים כירורגיים מדובר בבעיה קשה במיוחד עקב מורכבות הסצנה בניתוחים. בשנים האחרונות, הודות לשיפורים משמעותיים בחומרה (עם דגש על מאיצים גרפיים) וגישה לכמות רבה של נתונים (תמונות), נעשה שימוש בלמידה עמוקה לצורך פתרון בעיות מסוג זה. כיום, קיימת קשת נרחבת של ארכיטקטורות לרשת עמוקה. בעבודה זו אתמקד בארכיטקטורות מבוססות FCN (Fully Convolutional Networks) אשר הוצעו כפתרון לבעיה בתחרות שהתקיימה בשנת 2017 בסדנת EndoVis [A].

בחלק הראשון אגדיר ואציג את הבעיה כפי שהוצגה בתחרות, ואראה אילו אתגרים ייחודיים קיימים בבעיה זו. לאחר מכן, אציג ארכיטקטורות שונות אשר הוצעו כפתרון לבעיה, ואתמקד ואשווה בין פתרון המבוסס על ארכיטקטורת TernausNet [3] אשר ניצח בתחרות, לפתרון אלטרנטיבי ומהיר יחסית המבוסס על ארכיטקטורת LinkNet [4] שתוכנן גם הוא ע"י הצוות המנצח. לבסוף, אציג את התוצאות שהושגו באמצעות שתי הארכיטקטורות הללו ואת מסקנותיי.

# **סגמנטציה של מכשירים כירורגיים**

סגמנטציה של מכשירים כירורגיים זו בעיה חשובה אשר מהווה שלב ביניים במערכות ממוחשבות שמטרתן לעזור למנתח בביצוע ניתוחים זעיר פולשניים. בבעיה זו יש קושי מיוחד ביחס לסגמנטציה רגילה של תמונה, עקב הצורך בביצועה בזמן אמת. כלומר, יש לכלול את מיקום חלקי המכשור השונים באיבר המנותח כפונקציה של הזמן.

בשנת 2015 הוצע אתגר בתחום זה בסדנת EndoVis, אך הנתונים שהוצעו לא היו מספיק מגוונים והכילו טעויות אנוטציה [1]. בשנת 2017 הוצע אתגר נוסף שכלל 10 צוותים שהתמודדו עם בעיות סגמנטציה שונות של תמונות שנאספו מניתוחי בטן שונים אשר בוצעו על חזירים באמצעות מערכת da Vinci Xi של חברת Intuitive Surgical [I] – מדובר במערכת רובוטית מורכבת ויקרה אשר מספקת למנתח מגוון מכשירים כירורגיים (ראה איור 1). מערכת זו נמצאת בשימוש נרחב יחסית ותועלתה הוכחה עבור ניתוחים לפרוסקופיים שונים [2].



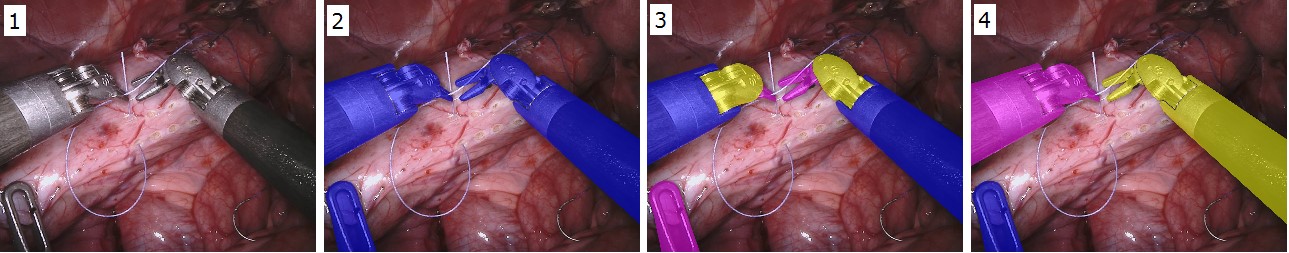
**איור 1** המכשירים הכירורגיים השונים של מערכת da Vinci Xi [1].

## **הגדרת הבעיה**

סגמנטציה של תמונה עוזרת לפשט את ניתוח התמונה ע"י חלוקתה לחלקים בעלי משמעות, ובכך לאפשר למחשב "להבין" את הסצנה.

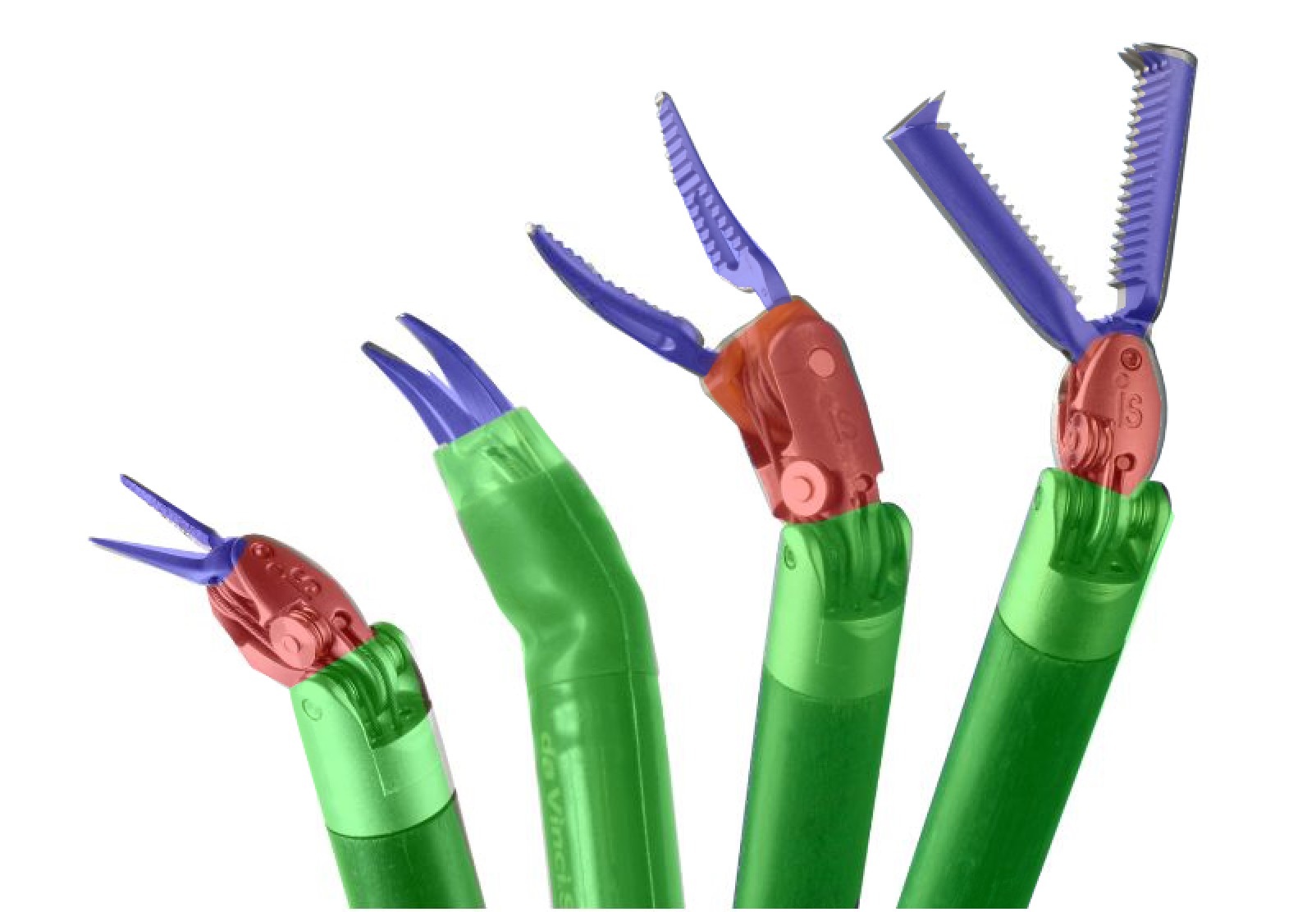
להלן בעיות הסגמנטציה השונות שהצוותים התמודדו איתן (ראה איור 2):

1. סגמנטציה בינארית – סגמנטציה של התמונה לשני חלקים: רקע ומכשירים.
2. סגמנטציה לפי חלקים – סגמנטציה של כל מכשיר על חלקיו השונים (ראה איור 3).
3. סגמנטציה לפי סוגים – סגמנטציה של המכשירים וסיווגם בהתאם.



**איור 2** בעיות הסגמנטציה השונות [2]: (1) התמונה המקורית, (2) סגמנטציה בינארית, (3) סגמנטציה לפי חלקים, (4) סגמנטציה לפי סוגים. יש לשים לב כי חלק הרקע אינו צבוע.

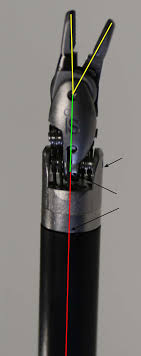
בכל אחת מהבעיות הנ"ל מקבלים תמונה בתור קלט, והמטרה היא להחזיר פלט אשר מורכב ממטריצת תוויות (מספרים) מתאימה. למשל, אם מדובר בסגמנטציה בינארית והפיקסל בתמונה הוא חלק ממכשיר כלשהו, אז יש להחזיר בתא של מטריצת התוויות את המספר המתאים לתווית המייצגת את המכשירים, ואם לא אז יש להחזיר את המספר המתאים לתווית המייצגת את הרקע. בהינתן מטריצת התוויות, ניתן לצבוע את החלקים השונים בצבעים שונים כפי שניתן לראות באיור 2.



**איור 3** ניתן לראות את החלקים השונים של המכשירים (מוטות בירוק, מפרקים באדום ואוחזים בכחול) [1].

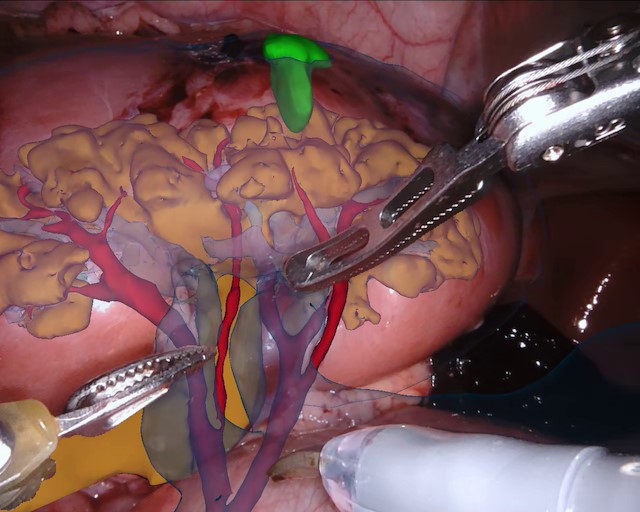
## **מוטיבציה**

ישנם יישומים פוטנציאליים רבים אשר תלויים בהבנת סצנת הניתוח [1]. סגמנטציה מדויקת של המכשירים הכירורגיים מהווה רכיב חיוני ביישומים מסוימים מסוג זה.



**איור 4** הערכת פוזה של מכשיר [5]. סגמנטציה לפי חלקים יכולה לשמש כשלב מקדים בביצוע הערכה של הטרנספורמציות השונות של צירי המכשיר. כמו כן, ניתן להשתמש בסגמנטציה לפי סוגים כדי לברר אילו פעולות המכשיר מסוגל לבצע בהתאם לסוגו ומיקומו המדויק בסצנה.

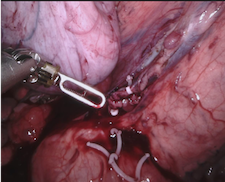
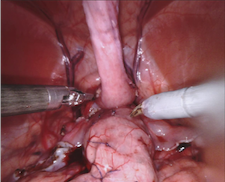
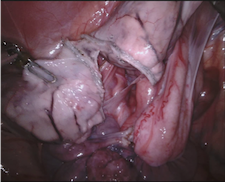
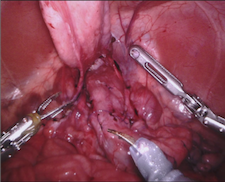
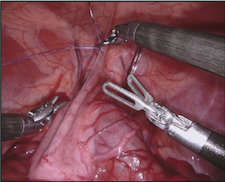
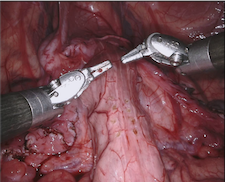
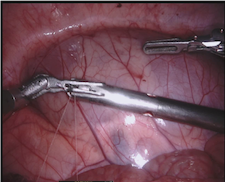
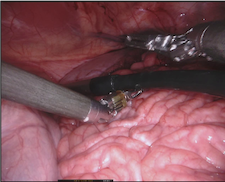
שימוש אפשרי וחשוב הוא ביצוע מעקב והערכת פוזה (ראה איור 4) של המכשירים לצורך אוטומציה של פעולות מסוימות ו\או עזרה למנתח בביצוען [5]. יישום אפשרי נוסף הוא מיסוך המכשירים בניתוחים במציאות רבודה (ראה איור 5) [1] – שיטה זו מאפשרת למנתח להתאמן ולתכנן את הניתוח האמיתי, או אפילו להשתמש בטכניקה כזו בזמן אמת.



**איור 5** מיסוך של המכשירים הכירורגיים כך שניתן יהיה לראותם במציאות הרבודה [1]. ניתן לעשות זאת ע"י שימוש בסגמנטציה בינארית.

## **הנתונים**

באתגר הוכנו 10 סדרות של תמונות של ניתוחי בטן שונים אשר בוצעו על חזירים באמצעות מערכת da Vinci Xi, וצוות ייעודי מ- Intuitive Surgical ביצע סגמנטציה ידנית שלהן. מכל צילום של ניתוח נלקחו קטעים רצופים שבהם נצפתה פעילות\תנועה משמעותית של המכשירים הכירורגיים, ומתוך קטעים אלו נדגמו באופן רציף 300 תמונות שונות ברזולוציה גבוהה (הצילום נדגם בקצב נמוך כדי להימנע מחזרה על תמונות זהות).



**איור 6** דוגמאות של תמונות מתוך הסדרות השונות [1].

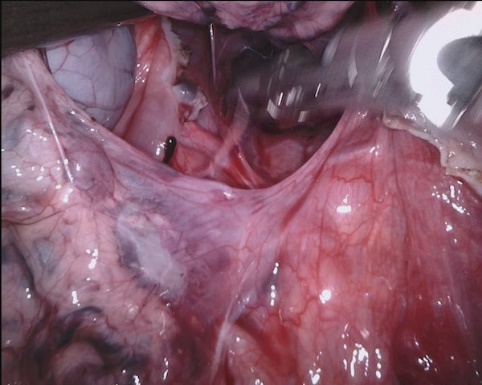
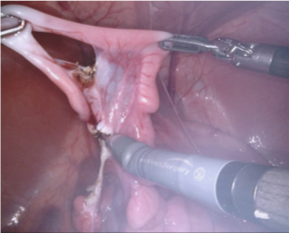
כל מקבץ של 300 תמונות רצופות מתוך 8 סדרות (ראה איור 6) חולק לקבוצת אימון שכללה את 225 התמונות הראשונות, וקבוצת מבחן שכללה את 75 התמונות האחרונות. שני המקבצים שנותרו שימשו כקבוצות מבחן גדולות נוספות. כלומר, עמדו לרשות כל מתמודד תמונות עם אנוטציה מתאימה, ובחנו כל אחד מהם על 10 קבוצות מבחן שונות ( תמונות).

לפי חוקי התחרות כל מודל שנבחן על קבוצת מבחן מסוימת מתוך 8 קבוצות המבחן הקטנות, לא היה יכול להתאמן על קבוצת האימון המתאימה לה מכיוון שהתמונות נלקחו מאותו ניתוח. הרי יש דמיון משמעותי בין תמונות שנלקחו מאותו ניתוח, ולכן אם מודל למד מקבוצת אימון מסוימת אז ביצועיו על קבוצת המבחן המתאימה לה לא בהכרח ישקפו את יכולת ההכללה של המודל. מפאת זאת, כל צוות היה צריך לאמן לפחות 9 מודלים שונים [1]. חל גם איסור על שימוש בתמונות נוספות, אך ניתן היה להשתמש ברשתות CNN (Convolutional Neural Network) אשר אומנו מראש על תמונות שאינן קשורות לניתוחים. כלומר, היה מותר לעשות שימוש בהעברת למידה (transfer learning) עקב הכמות הקטנה של התמונות בקבוצות האימון.

יש לציין כי רוב הצוותים עשו שימוש בטכניקות שונות של עיבוי נתונים (data augmentation) – ארחיב על כך בהמשך.

## **אתגרים**

סגמנטציה של תמונה היא בעיה כללית ומוכרת בראייה ממוחשבת וכרוכים בה אתגרים מסוימים בנוסף לאתגרים הקיימים בבעיות סיווג רגילות. למשל, יש קושי מיוחד בביצוע אנוטציה מכיוון שיש צורך לסמן באופן מדויק היכן נמצא כל חלק. כלומר, צריך להגדיר תווית מתאימה עבור כל פיקסל. ישנן תוכנות המאפשרות לעשות זאת ביתר קלות, אך עדיין מדובר בעבודה ידנית רבה המועדת לטעויות [1]. כמו כן, לפעמים לא לגמרי ברור איך נכון לחלק את התמונה (ראה איור 7). כלומר, יש מרכיב סובייקטיבי שיש לתת עליו את הדעת [1]. קושי משמעותי נוסף נובע מכך שהפלט המבוקש הוא תמונה בגודל זהה לתמונה המקורית אשר מורכבת מאלפים רבים ואף מיליונים בודדים של פיקסלים, ומכאן שמודלים הפותרים בעיה זו הם מורכבים ואיטיים יחסית. עקב כך, זמן הריצה עלול להוות מגבלה משמעותית במקרה שיש צורך בסגמנטציה בזמן אמת.



**איור 7** המכשיר בתמונה הימנית מטושטש ולכן לא ברור איפה נמצאים קווי המתאר שלו. בתמונה השמאלית ניתן לראות את העשן שנוצר מצריבת רקמות [1]. קל גם להבחין בהחזרי ראי רבים על המכשירים והרקמות.

ישנם אתגרים ייחודיים במקרה של סגמנטציה של מכשירים כירורגיים:

* לרוב יש צורך בביצוע הסגמנטציה בזמן אמת.
* סצנת הניתוח מאוד מגוונת (יכולים להיות הבדלים אנטומיים משמעותיים בין אנשים שונים) ולא מספיק נגישה ולכן קשה לאסוף מערכי נתונים גדולים ומגוונים, ומכאן שלא קיימים מערכים כאלו (בניגוד לתמונות שגורות שעושים בהן שימוש בהרבה בעיות של ראייה ממוחשבת: תמונות של בני-אדם, בעלי חיים, חפצים, וכו'). לכן, קשה להסיק לגבי היתרונות והחסרונות של אלגוריתמים אשר פותרים את בעיה זו, ולא ניתן להפיק תועלת מיוחדת משימוש ברשתות מורכבות במיוחד (תתכן גם התאמת יתר). לצורך השוואה, בתחרות יש רק 1800 תמונות בקבוצות האימון לעומת מאות אלפים במקרה של מערכי נתונים סטנדרטיים של ראייה ממוחשבת [1].
* סצנת הניתוח די ייחודית ושונה משמעותית מסצנות שגורות ולכן קשה לדעת אילו טכניקות של עיבוי נתונים תהיינה מועילות [1]. יתר על כן, יתכן כי טכניקות מסוימות יגרמו לפגיעה בביצועי המודל במקרה שתוצאותיהן אינן מאפיינות את השינויים המתרחשים בפועל [1].
* ישנם שינויים משמעותיים של צללים והחזרי ראי על המכשירים והרקמות הנגרמים ממקור האור, רקמות שונות ומורכבות ברקע, חסימות\הפרעות בשדה הראייה הנגרמות מדם או מעשן שנוצר מצריבת רקמות, טשטוש שנוצר מהזזה מהירה של המכשירים, וכו' (ראה איור 7) [1].
* התפלגות הופעתם של המכשירים השונים בנתונים אינה מייצגת בהכרח את המציאות עקב הגודל המזערי של קבוצת האימון [1].

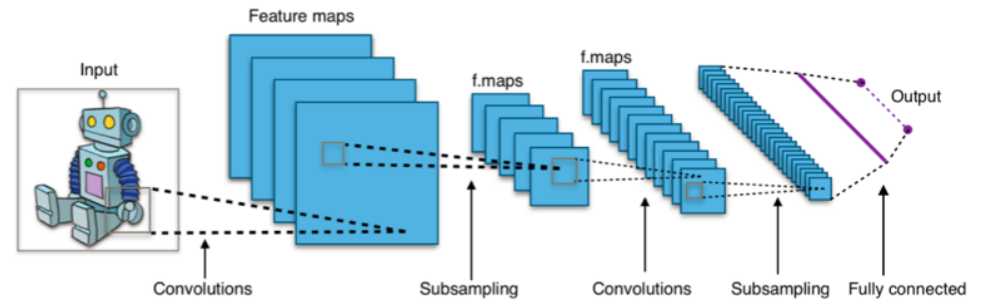
# **ארכיטקטורות מבוססות FCN**

בשנים האחרונות, הודות לשיפורים משמעותיים בחומרה (עם דגש על מאיצים גרפיים) וגישה לכמות רבה של נתונים (תמונות), נעשה שימוש ברשתות עמוקות לצורך פתרון מגוון רחב של בעיות בתחום הראייה הממוחשבת (סיווג תמונות, איתור אובייקטים, הבנת סצנה, וכו'). רשתות אלו הוכיחו את עליונותן על שיטות מסורתיות של למידת מכונה, ואף על בני-אדם המומחים למקצועות הרלוונטיים [3].

בפרק מרכזי זה אציג תחילה את המבנה הכללי של רשתות CNN (Convolutional Neural Network) המהווה את הבסיס של המבנה הכללי של רשתות FCN (Fully Convolutional Network) אשר אותו אציג מיד לאחר מכן. לבסוף, אציג את הארכיטקטורות השונות המבוססות על מבנה כללי זה אשר הוצעו כפתרון לבעיות הסגמנטציה של מכשירים כירורגיים בתחרות EndoVis 2017.

## **רשתות CNN**

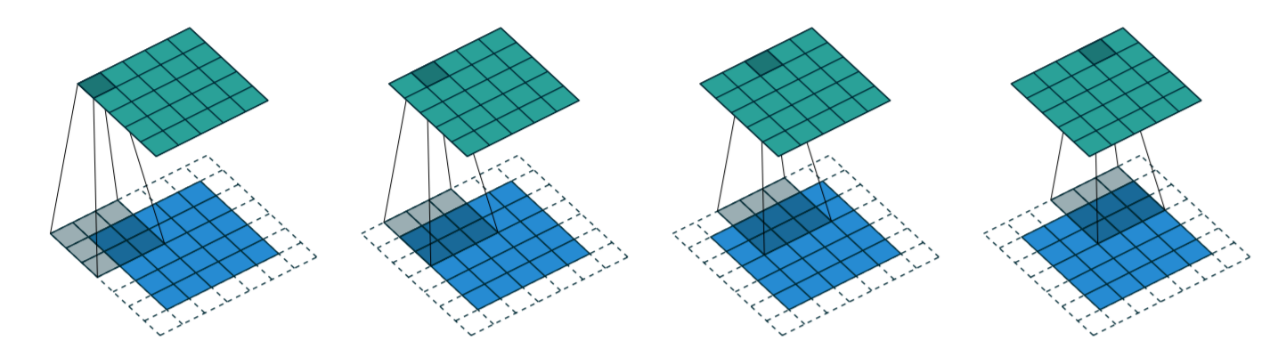
אחד הקשיים העיקריים בלמידה מתמונות הוא מספר הממדים הרב של הנתונים. כלומר, כל תמונה מורכבת מהרבה ערכי צבע של פיקסלים – עובדה זו מחייבת שימוש בכמות רבה מאוד של נתונים לצורך השגת יכולת הכללה גבוהה של המערכת הלומדת. בנוסף, יש צורך בכוח עיבוד וכמות זיכרון משמעותיים הן בשלב הלמידה והן בשלב ההסקה (inference). רשת CNN זו רשת זרימה קדימה (feedforward neural network) אשר מהווה את אחת השיטות המוכרות והמוצלחות ביותר להתמודדות עם אתגרים אלו (ראה איור 8).



**איור 8** אילוסטרציה של רשת CNN [J]. בהינתן מפות מאפיינים (feature maps) בשכבה הנוכחית (או ערוצי RGB של התמונה בשכבה הראשונה), מחשבים מפת מאפיינים עבור כל ערוץ (channel) בשכבה הבאה באופן הבא: מבצעים פעולות קונבולוציה על המפות של השכבה הנוכחית עם פילטרים (kernels\filters – ריבועים אפורים באיור), מחברים את תוצאות אלו (וערך bias), ותוצאה זו (לאחר הפעלת פונקציית אקטיבציה) מהווה את אחת ממפות המאפיינים של השכבה הבאה. לבסוף, משטחים (flattening) את מפות המאפיינים בשכבת המפות האחרונה לכדי וקטור מאפיינים אחד (קו סגול באיור), ומכאן ממשיכים עם שכבות רגילות המחוברות באופן מלא (fully connected) בדומה לרשת נוירונים רגילה. מדי פעם משתמשים בפעולות הקטנת רזולוציה (subsampling). בין השכבות של המפות. במקרה של סיווג תמונות, מספר הנוירונים בשכבה האחרונה שווה למספר המחלקות (עם אקטיבציה Softmax לצורך קבלת התפלגות של הסתברויות עבור המחלקות השונות).

ההבדלים המהותיים בין רשת נוירונים רגילה לבין רשת נוירונים קונבולוציונית הם כדלקמן:

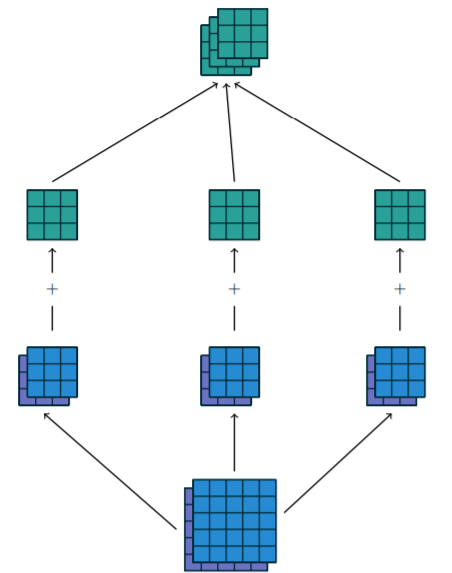
1. ישנן שכבות של מפות מאפיינים במקום נוירונים בודדים (מלבד השכבות האחרונות), כאשר מספר הערוצים מגדיר את מספר המפות בשכבה.
2. מתבצעות פעולות קונבולוציה על מפות עם מטריצות משקלים קטנות שנקראות פילטרים (kernels\filters), במקום פעולות כפל רגילות של ערכי נוירונים במשקלים (ראה איורים 9 ו- 10).
3. מתבצעות פעולות חוזרות ונשנות של downsampling\subsampling (הקטנת רזולוציה) על המפות תוך הגדלה של מספר הערוצים בשכבות העמוקות. שיטה פשוטה ומועילה לביצוע downsampling נקראת Max pooling (ראה איור 11).



**איור 9** אילוסטרציה של פעולת הקונבולוציה [12]. בהינתן מפת מאפיינים (כחול) ו- kernel (אפור), מתבצעת פעולת הקונבולוציה באופן הבא (בסדר משמאל לימין, מלמעלה למטה): משקלי ה- kernel מוכפלים בערכים המתאימים של המפה, מתבצע חיבור של ערכים אלו, ותוצאה זו מהווה את הערך הבא של המפה הנוצרת (ירוק). ניתן להשתמש בגדלי kernel שונים כדי לשלוט במספר הפרמטרים ובשדה הקליטה (receptive field) של ה- kernel. כמו כן, ניתן להשתמש בריפוד (padding – ריבועים מקווקוים מאופסים באיור) ובגדלי צעד (stride) גדולים מ- 1 כדי לשלוט ברזולוציה של המפה הנוצרת. יש להדגיש כי משקלי ה- kernel בד"כ משותפים לכל הצעדים מכיוון שמאפיין שימושי מסוים יכול להופיע במקומות שונים בתמונה.

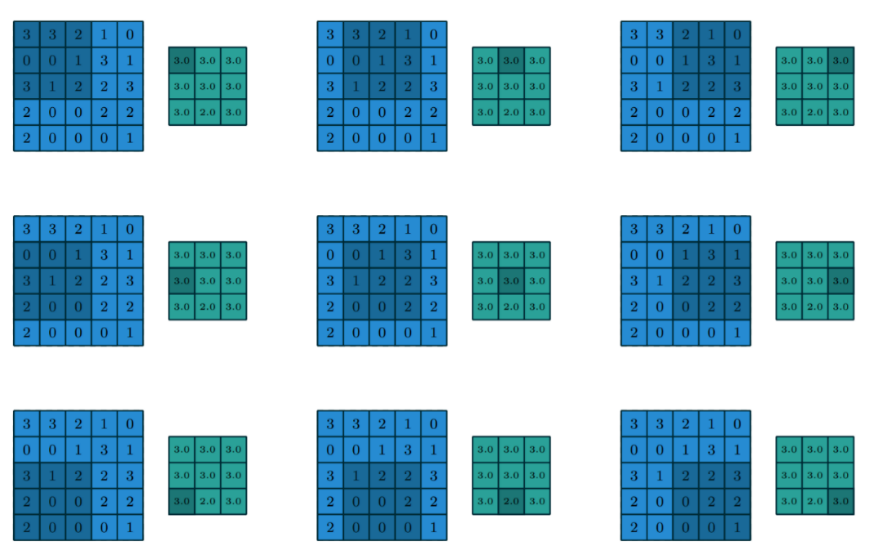
השילוב של שלושת ההבדלים הנ"ל מאפשר ניצול של הקשר המרחבי הקיים בין פיקסלים שכנים\קרובים כך שהמפות הנוצרות בעצם מהוות מפות של מאפיינים מרחביים (spatial feature maps) המסודרות במבנה היררכי של רמות אבסטרקציה שונות. כלומר, המפות בשכבות הרדודות מכילות מאפיינים מרחביים פשוטים כגון קווים ועיגולים, והמפות בשכבות העמוקות מכילות מאפיינים מורכבים כגון חלקים שלמים של אובייקט. לכן, יש צורך במספר ערוצים רב יותר בשכבות העמוקות. מבנה כללי זה מאפשר לרשתות CNN להשיג יכולת הכללה גבוהה גם במקרה של קבוצת אימון מצומצמת יחסית.

פעולות ה- downsampling משמשות כפעולות הורדת ממד החוסכות בזמן ריצה וזיכרון, תוך פעפוע קדימה של מאפיינים מרחביים שנמצאו במיקומים שונים על המפות של השכבה הקודמת. כלומר, בשכבות העמוקות מתקבל מידע סמנטי לגבי האובייקטים הקיימים בתמונה, אך מידע לגבי מיקומם המדויק נשאר בשכבות הרדודות – עובדה זו אינה מהווה בעיה במקרה של סיווג תמונה, אך ברור כי בבעיית הסגמנטציה יש למידע הנ"ל חשיבות רבה.



**איור 10** דוגמה לחישוב של מפות המאפיינים בשכבה הבאה בהינתן המפות של השכבה הנוכחית [12]. בשכבה הנוכחית (כחול) יש שני ערוצים ובשכבה הבאה (ירוק) יש שלושה ערוצים, ולכן ישנו זוג פילטרים עבור כל ערוץ בשכבה הבאה. המפות הכחולות הקטנות מתקבלות ע"י ביצוע קונבולוציה על המפות הכחולות בשכבה הנוכחית עם זוגות הפילטרים הנ"ל. לאחר מכן, מתבצע חיבור של תוצאות אלו. לבסוף, מחברים למפות הנוצרות ערכי bias (לכל ערך במפה מחובר ערך bias זהה. כלומר, יש סה"כ שלושה ערכי bias שונים) ומפעילים את פונקציית האקטיבציה (חסר באיור).

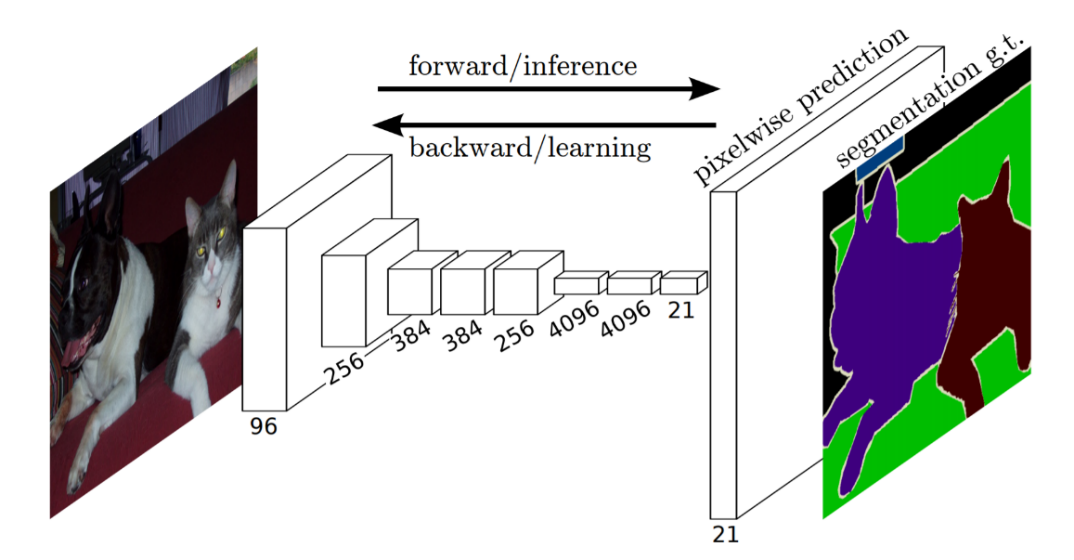
בדומה לרשתות נוירונים רגילות, הלמידה מתבצעת ע"י חישוב הגרדיאנט של פונקציית ההפסד לפי משקלי הרשת באמצעות אלגוריתמי פעפוע לאחור (backpropagation), ועדכון שלהם באמצעות אלגוריתמי למידת גרדיאנט (gradient descent) המביאים (בתקווה) למזעור של שגיאת ההכללה.



**איור 11** אילוסטרציה של פעולת Max pooling עם פילטר בגודל 3x3 ו- stride בגודל 1 [12]. לאחר ביצוע קונבולוציות, ניתן להקטין את רזולוציית המפה (downsampling) ע"י ביצוע מעבר עם פילטר (בדומה לפעולת הקונבולוציה) שפשוט בוחר את הערך המקסימלי מבין הערכים הנוכחיים. בניגוד לשימוש בקונבולוציה עם stride גדול מ- 1, פעולה זו מוגדרת מראש והרשת לא צריכה ללמוד לבצעה באופן מועיל. יש לציין כי בד"כ משתמשים בפילטר בגודל 2x2 ו- stride בגודל 2 כדי להקטין את הרזולוציה באופן משמעותי.

## **רשתות FCN**

אחת השיטות המוצלחות ביותר לביצוע סגמנטציה של תמונה היא שיטה המבוססת על ארכיטקטורת FCN (Fully Convolutional Network) [7] (ראה איור 12). שיטה זו עושה שימוש במבנה הפשוט והמוכר של רשתות CNN ללא השכבות האחרונות המחוברות באופן מלא (fully connected layers), בתור חולץ של מפות מאפיינים מרחביים. כלומר, בתור החולץ ניתן להשתמש ברשת CNN שאומנה מראש ע"י הסרת חלק משכבותיה האחרונות. לאחר מכן, מתבצעות פעולות הגדלת רזולוציה (upsampling) על מפות אלו לצורך הפקת מפות מאפיינים ברזולוציה גבוהה יותר המשמשות לקבלת פרדיקציה ברמת הפיקסלים. פרדיקציה זו משמשת לקביעת הסיווגים של הפיקסלים ובכך מתקבלת תוצאת הסגמנטציה.



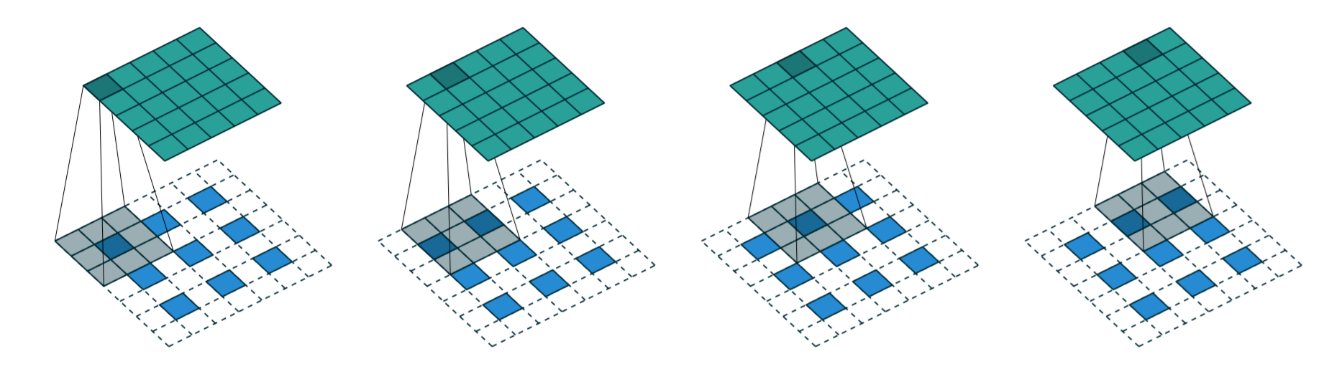
**איור 12** המבנה הכללי של רשת FCN [7]. בדומה לרשת CNN, תחילה מתבצעות פעולות קונבולוציה ו- downsampling (באמצעות Max pooling למשל) אשר מקטינות את רזולוציית מפות המאפיינים תוך הגדלה של מספר הערוצים (מספר הערוצים מופיע מתחת למפות) בשכבות העמוקות. במקום ביצוע השטחה של המפות בשכבה האחרונה, מתבצעות עליהן פעולות upsampling המפיקות מפות מאפיינים ברזולוציה גבוהה המשמשות לקבלת פרדיקציה ברמת הפיקסלים. כלומר, בשכבה האחרונה יש אקטיבציה Softmax ומספר הערוצים שווה למספר המחלקות.

כפי שניתן לראות באיור 12, הרישא של הרשת היא בעלת מבנה של רשת CNN רגילה. כלומר, ישנן שכבות קונבולוציה במבנה היררכי תוך הקטנה של רזולוציה והגדלה במספר הערוצים בשכבות העמוקות. בניגוד לרשת CNN, לא מתבצעת השטחה של המפות בשכבה האחרונה, אלא מתבצעות פעולות upsampling כפי שכבר ציינתי (רזולוציית המפות בשכבה האחרונה שווה לרזולוציה של התמונה). ישנן שיטות שונות לביצוע פעולת ה- upsampling מלבד שיטות פשוטות כדוגמת אינטרפולציה בליניארית. שיטה מוצלחת במיוחד היא פעולת הקונבולוציה המשוחלפת (ראה איור 13).

יש לשים לב כי מספר הערוצים בשכבה האחרונה שווה למספר המחלקות הנתונות בבעיית הסגמנטציה המבוקשת (יש 21 מחלקות בבעיה הרלוונטית לאיור 12) כאשר ישנה אקטיבציה Softmax ברמת הפיקסלים לצורך קבלת התפלגות של הסתברויות בכל פיקסל בהתאם למספר המחלקות. כלומר, ההסתברות שפיקסל שייך למחלקה נתונה ע"י:

כאשר שווה לערך האקטיבציה במיקום של המפה בשכבת הפרדיקציה, ו- שווה למספר המחלקות. לכן, ניתן לקבוע את סיווג הפיקסל בהתאם ל:

כלומר, החלטת הסיווג מתבצעת בכל פיקסל בנפרד בדומה לאופן שבו הדבר נעשה בשכבה האחרונה של רשת CNN רגילה, ומכאן מתקבלת תוצאת הסגמנטציה הדרושה (ראה תוצאה סופית רצויה בצד ימין של איור 12).



**איור 13** אילוסטרציה של פעולת קונבולוציה משוחלפת (transposed convolution\deconvolution) עם stride בגודל 2 [12]. פעולת הקונבולוציה המשוחלפת משמשת כטרנספורמציה המבצעת upsampling לקלט, וניתן להבין את אופן פעולתה כפעולת קונבולוציה רגילה עם stride בגודל , ולכן היא מוכרת גם בשם fractionally strided convolution [12]. כלומר, מוכנסות שורות ועמודות אפסים בין ערכי הקלט ועל תוצאה זו מבוצעת הקונבולוציה, ומכאן שרזולוציית הפלט גדלה בהתאם. היתרון המרכזי של פעולה זו כשיטה לביצוע upsampling, נעוץ בכך שהיא מאפשרת לרשת ללמוד לבצע פעולות upsampling מורכבות ומועילות במיוחד [7].

אחד היתרונות המרכזיים של ארכיטקטורות המבוססות על המבנה הכללי הנ"ל הוא האפשרות של אימון הרשת מקצה לקצה (end-to-end) לצורך ביצוע סגמנטציה של תמונות ברזולוציה כלשהי [3]. יתרון זה מתאפשר באופן טבעי עקב אי-השימוש בשכבות המחוברות באופן מלא.

בתת-פרק הבא, אציג ארכיטקטורות מבוססות FCN אשר הוצעו כפתרון לבעיות הסגמנטציה של מכשירים כירורגיים בתחרות EndoVis 2017.

## **הארכיטקטורות השונות בתחרות EndoVis 2017**

בתחרות EndoVis 2017 התמודדו 10 צוותים מאוניברסיטאות ומכוני מחקר שונים מרחבי העולם. כל הצוותים הציעו פתרון המבוסס על רשת עמוקה, מלבד הצוות מאוניברסיטת וושינגטון שעשה שימוש בשיטות מסורתיות של ראייה ממוחשבת אשר אינן מבוססות על למידת מכונה.

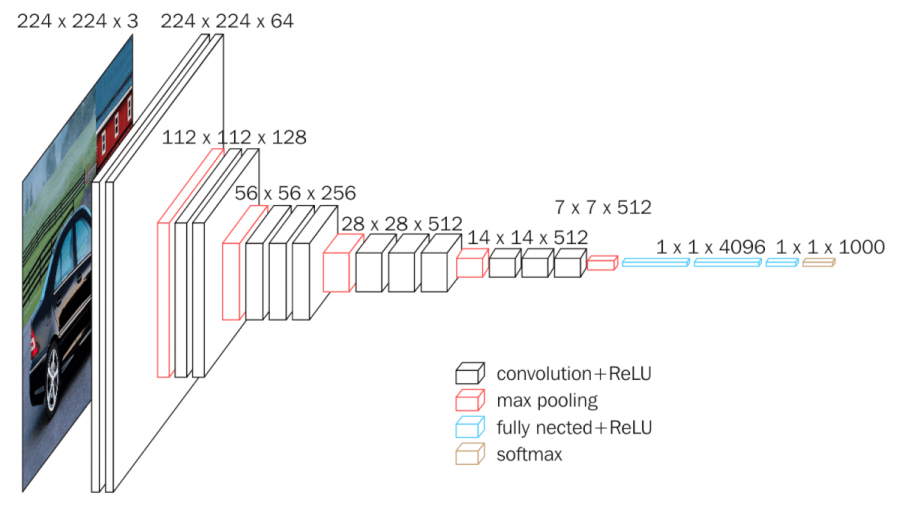
להלן רשימת הצוותים:

* המרכז הלאומי למחלות גידוליות, דרזדן (NCT)
* אוניברסיטת ברן (UB)
* מכון טכנולוגי בייג'ינג (BIT)
* מכון טכנולוגי מסצ'וסטס (MIT)
* מכון שנג'ן לטכנולוגיה מתקדמת (SIAT)
* מכללת אוניברסיטה לונדון (UCL)
* האוניברסיטה הטכנית, מינכן (TUM)
* מכון אינדרפראסטה לטכנולוגיית מידע, דלהי (IIITD)
* אוניברסיטת אלברטה (UA)
* אוניברסיטת וושינגטון (UW)

רק 6 צוותים (NCT, UB, MIT, SIAT, UCL, UA) הציעו פתרון שנתן מענה לכל אחת משלושת בעיות הסגמנטציה, ולכן אציג את פתרונות אלו בלבד. את הפתרון המנצח (MIT) אציג ביתר פירוט בתת-פרק הבא.

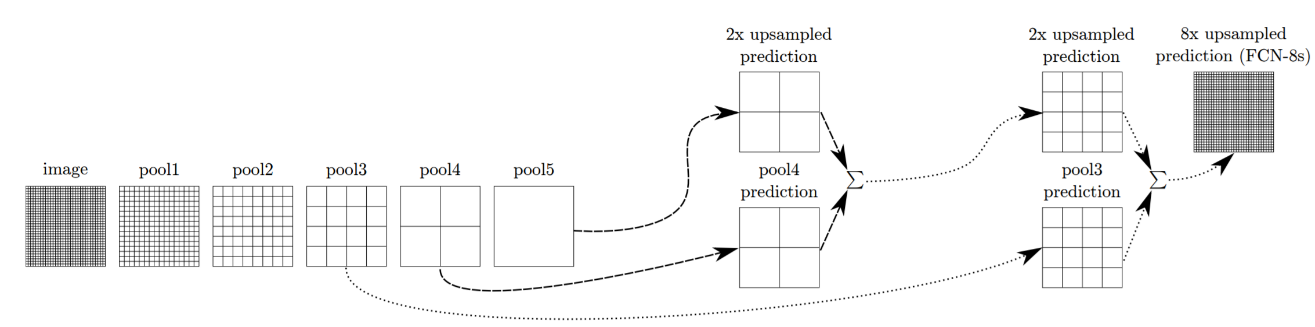
הפתרון של UA:

פתרון זה עושה שימוש ברשת המבוססת על ארכיטקטורה פשוטה יחסית שנקראת FCN-8 שאומנה מראש על התמונות של PASCAL [7, E] (מערך זה מכיל תמונות של אובייקטים שגורים עם אנוטציה לצורך ביצוע סגמנטציה של תמונות). רשת זו משתמשת ברשת CNN שנקראת VGG16 [C] (ראה איור 14) בתור חולץ מאפיינים שאומן מראש על התמונות של ImageNet [D] (מערך זה מכיל תמונות של אובייקטים שגורים עם אנוטציה לצורך ביצוע סיווג של תמונות).



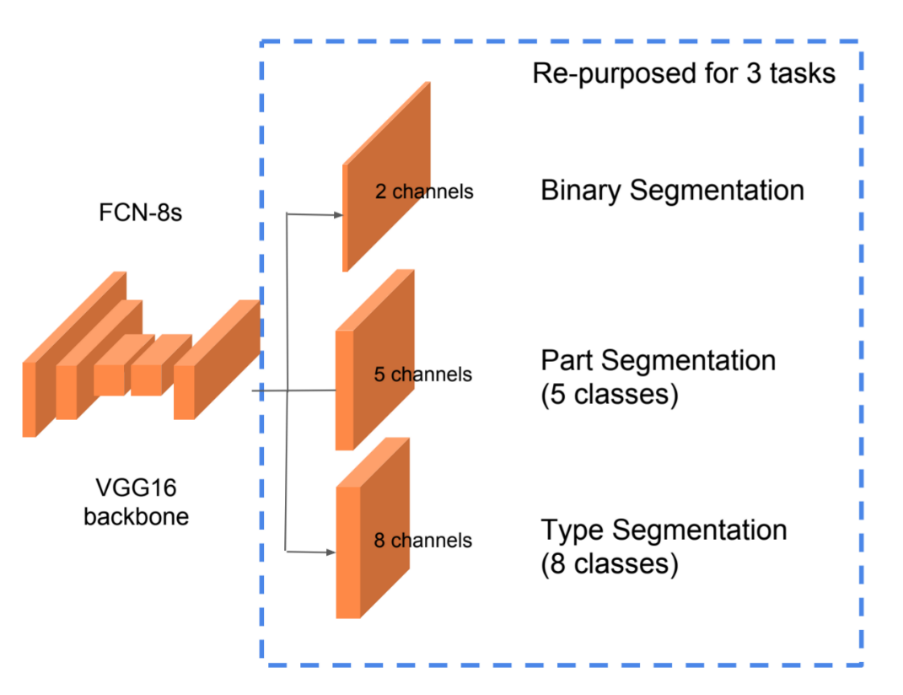
**איור 14** הטופולוגיה של ארכיטקטורת VGG16 [C]. מדובר ברשת CNN מוכרת אשר מבצעת סיווג של תמונות מעל 1000 מחלקות שונות. רשת זו משיגה תוצאות טובות על התמונות של ImageNet.

בגישה זו מבצעים upsampling (באמצעות קונבולוציה משוחלפת) על מפות מאפיינים ברזולוציה נמוכה אשר מכילות מידע סמנטי לגבי האובייקטים השונים בתמונה, ומחברים אותן למפות מאפיינים ברזולוציה גבוהה יותר אשר מכילות מידע הקשור למיקום של האובייקטים (ראה איור 15) [7]. שיטה זו מפיקה סגמנטציה מדויקת יותר מאשר ביצוע upsampling רק על השכבה העמוקה ביותר עקב השילוב של סמנטיקה ומיקום.



**איור 15**  ניתן לראות כיצד מתקבלת תוצאת הסגמנטציה ברשת FCN-8 [7]. מבצעים upsampling (x2) על מפה pool5 ומחברים אותה למפה pool4. לאחר מכן, מבצעים upsampling (x2) על המפה המתקבלת ומחברים אותה למפה pool3. לבסוף, מבצעים upsampling (x8) לקבלת מפה ברזולוציה הזהה לזו של התמונה המקורית. מפות אלו משמשות לחישוב הפרדיקציה של הפיקסלים.

כדי להתמודד בצורה טובה יותר עם שלושת בעיות הסגמנטציה, הצוות עשה שימוש בהעברת למידה עם כוונון עדין (fine-tuning) [B] – בטכניקה זו מקבעים את המשקלים של רישא של הרשת, ומאמנים את הסיפא שלה על הנתונים החדשים. השכבות הראשונות של הרשת המאומנת מראש למדו מאפיינים כלליים אשר רלוונטיים גם לנתונים החדשים, ומכאן ההיגיון בשיטה זו. יש לציין כי זמן האימון גם מתקצר כי חלק מהמשקלים לא צריכים להתעדכן.



**איור 16** אילוסטרציה כללית של הפתרון של UA [1].

תחילה אימנו מודלים הפותרים את בעיית הסגמנטציה הבינארית ע"י שימוש במשקלים של הרשת FCN-8 הנ"ל, והמשקלים של המודל בעל הביצועים הטובים ביותר שמשו כמשקלים התחלתיים עבור המודלים שאומנו בצורה דומה כדי לפתור את בעיות הסגמנטציה האחרות (ראה איור 16).

המודלים אומנו על תמונות ברזולוציה נמוכה כדי לקצר את זמן האימון, ולא נעשה שימוש בטכניקות של עיבוי נתונים.

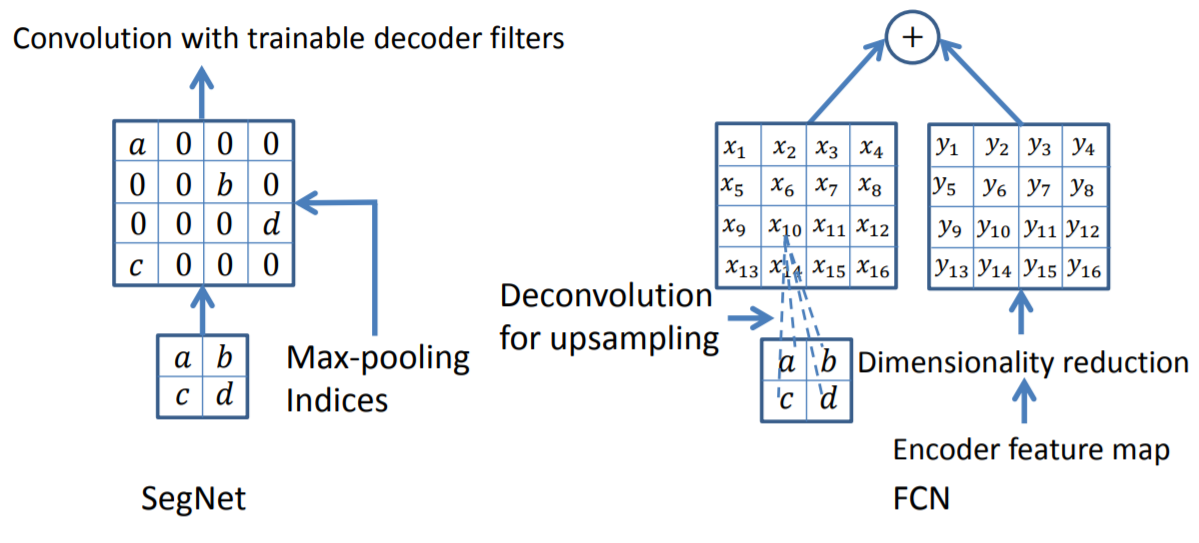
הפתרון של SIAT:

פתרון זה מבוסס על ארכיטקטורה שנקראת SegNet [8]. ארכיטקטורה זו היא בעלת מבנה של מקודד-מפענח (ראה איור 17) אשר עושה שימוש באינדקסים של תוצאת פעולות ה-Max pooling המתבצעות בחלק המקודד, לצורך ביצוע פעולות ה- upsampling בחלק המפענח (ראה איור 18). שיטה זו משפרת את היכולת של הרשת לזהות היכן נמצאים קווי המתאר של האובייקטים בתמונה, וגם מייתרת את הצורך ללמוד לבצע את פעולת ה- upsampling (אין צורך בקונבולוציה משוחלפת), ומכאן שיש גם חיסכון במספר הפרמטרים של הרשת [8].



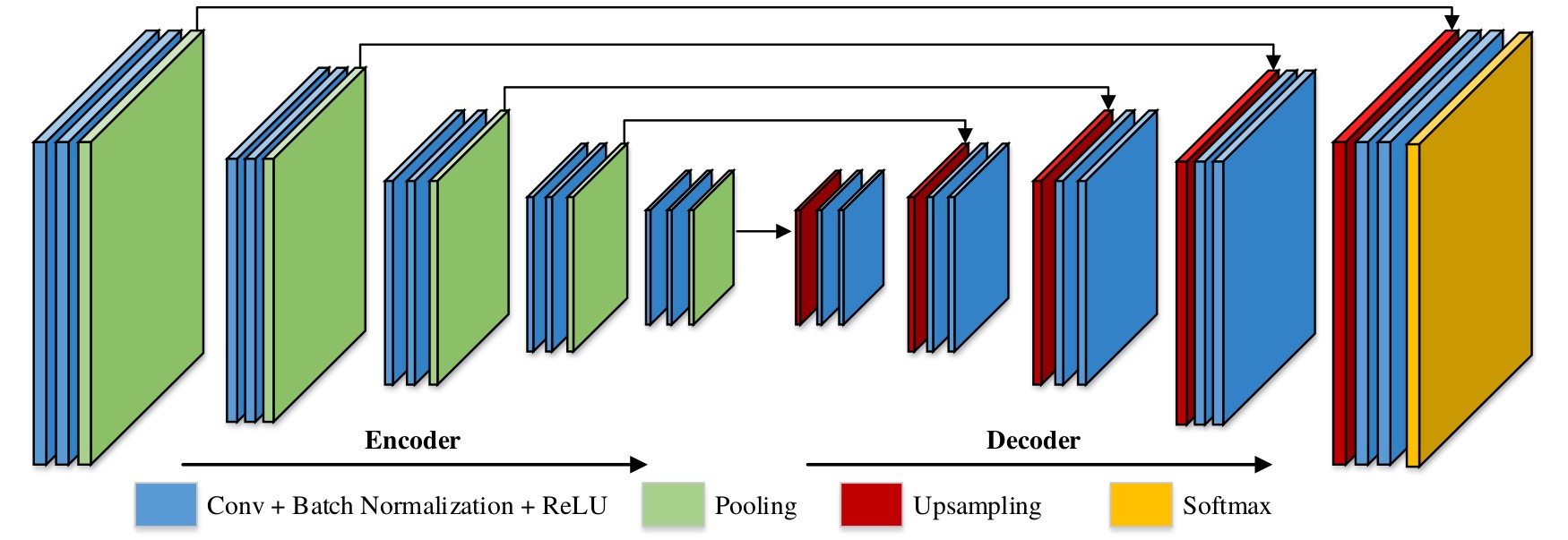
**איור 17** הטופולוגיה הכללית של ארכיטקטורת מקודד-מפענח [13]. החלק השמאלי "מקודד" את התמונה ע"י יצירת וקטור של מאפיינים (באופן דומה לרשת CNN רגילה), והחלק הימני "מפענח" את מאפיינים אלו ומחזיר את הפרדיקציה של הפיקסלים אשר משמשת לצורך הסגמנטציה.

בדומה לפתרון של UA, גם כאן נעשה שימוש בטכניקת העברת למידה עם כוונון עדין ע"י שימוש ברשת VGG16 כחולץ מאפיינים אשר אומן מראש על התמונות של ImageNet.



**איור 18** ניתן לראות את ההבדל בין שימוש באינדקסים של פעולת ה- Max pooling לבין שיטת ה- upsampling שנעשה בה שימוש בארכיטקטורת FCN-8 [8]. בצד שמאל ניתן לראות כי הערכים של מפת מאפיינים בחלק המפענח ממוקמים ע"י פעולת ה- upsampling בהתאם לאינדקסים של הארגומנטים בעלי הערך המקסימלי אשר התקבלו במפת המאפיינים המתאימה בחלק המקודד לאחר פעולת ה- Max pooling. בצד ימין ה- upsampling מתבצע באמצעות קונבולוציה משוחלפת, ולאחר מכן מתבצעת פעולת החיבור עם מפה מהשכבה הקודמת.

מודלים שונים של הפתרון (ראה איור 19) אומנו עבור שתי בעיות הסגמנטציה הראשונות – בינארית וחלקים – ומודלים נוספים אומנו עבור בעיית הסגמנטציה לפי סוגים ע"י שינוי מספר המחלקות בשכבה האחרונה בהתאם, וביצוע כוונון עדין של שכבה זו (העברת הלמידה בוצעה מבעיית הסגמנטציה לפי חלקים לבעיית הסגמנטציה לפי סוגים).



**איור 19** הטופולוגיה של הפתרון של SIAT [1]. החיצים מייצגים את מעבר המידע לגבי האינדקסים המתאימים לתוצאות פעולת ה- Max pooling לצורך ביצוע פעולת ה- upsampling. החלק המקודד מורכב מרישא של רשת VGG16 שאומנה מראש על התמונות של ImageNet.

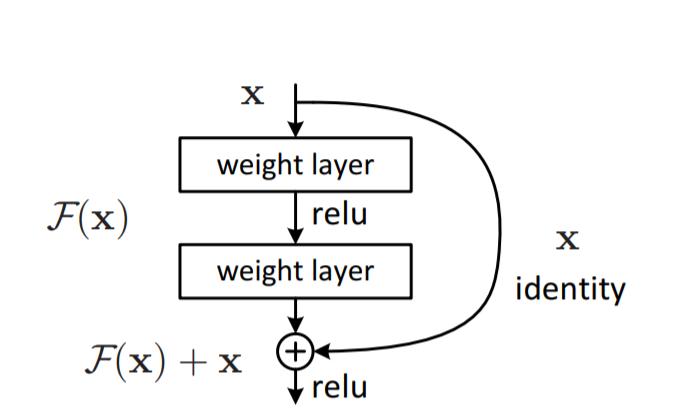
לאחר פעולות הקונבולוציה נעשה שימוש בנרמול אצווה (Batch normalization) [11] – מדובר בפעולת נרמול מוכרת ומוצלחת במיוחד אשר מקצרת משמעותית את זמן האימון ועוזרת במניעת התאמת יתר.

ניואנס נוסף אשר הצוות נתן עליו את הדעת בתכנון הפתרון, הוא חוסר האיזון בהתפלגות הופעתם של חלקי המכשירים בנתונים. לצורך כך, נעשה שימוש במשקלים בפונקציית ה- Softmax של השכבה האחרונה כדי לרכך את חוסר האיזון.

המודלים אומנו על תמונות ברזולוציה נמוכה עקב מגבלות של חומרה (כוח עיבוד\זיכרון), ונעשה שימוש בטכניקות של עיבוי נתונים: היפוך, שינוי צבע, ניגודיות, חדות.

הפתרון של NCT:

פתרון זה מבוסס על מבנה מקודד-מפענח עם חיבורים שיוריים (residual connections) ארוכים בדומה לרשת קונבולוציונית שיורית (residual CNN) [10] – רשתות מסוג זה מורכבות מרכיבים שנקראים בלוקים שיוריים (residual blocks) כפי שניתן לראות באיור 20.

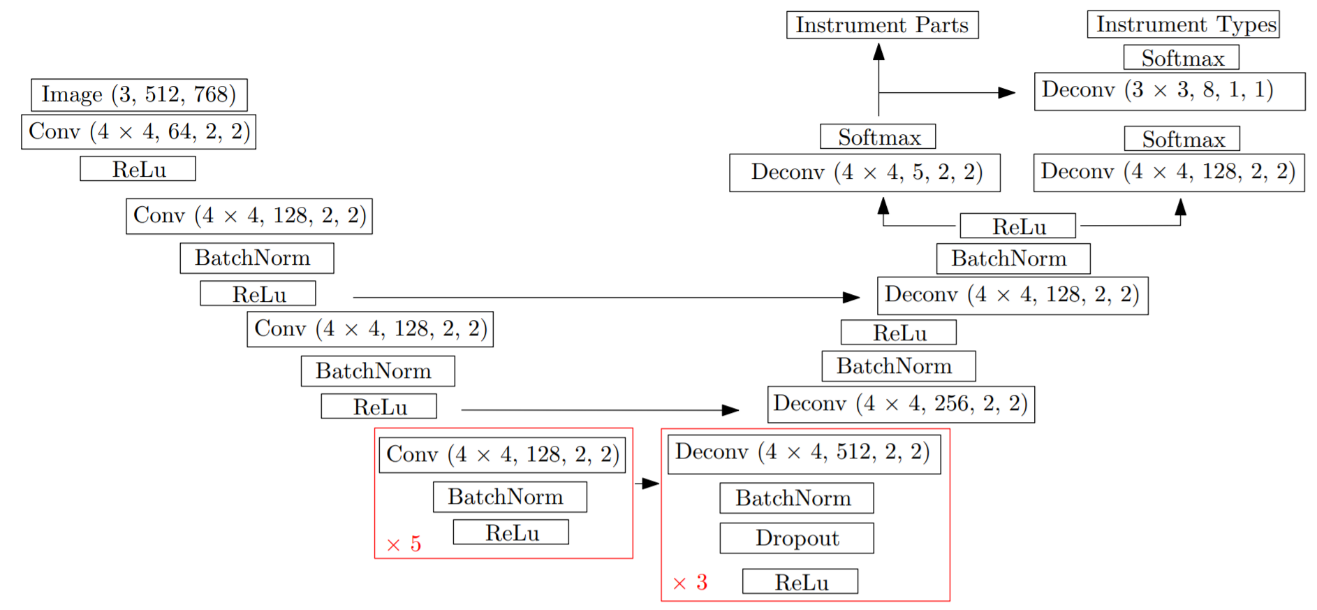


**איור 20** הבלוק השיורי (residual block) [10]. בלוקים מצורה זו מהווים את אבני הבניין של רשתות שיוריות. כל בלוק מורכב ממספר שכבות וחיבור אשר מחבר את הקלט של השכבה הראשונה של הבלוק, עם הפלט של השכבה האחרונה שלו. שיטה זו עוזרת להתמודד עם בעיית הדגרדציה (degradation) של הדיוק המתרחשת בשלב האימון. בעיה זו נוטה לקרות ברשתות עמוקות במיוחד עקב מורכבות הפונקציה שהרשת אמורה לקרב [10].

ניתן לראות את הטופולוגיה של הרשת באיור 21. באופן כללי, הרשת מבצעת קונבולוציות עם פונקציית אקטיבציה ReLU ו- stride בגודל 2 לצורך ביצוע downsampling. לאחר ביצוע פעולות הקונבולוציה, מתבצעות קונבולוציות משוחלפות. כמו כן, נעשה שימוש חוזר ונשנה בנרמול אצווה ו- Dropout – פעולה זו עוזרת למנוע התאמת יתר של המודל ע"י ביטול אקראי של נוירונים תוך כדי האימון, אך לא ברור אם שיטה זו כדאית כאשר היא מופעלת אחרי שכבת קונבולוציה [F]. לבסוף, נעשה שימוש בפונקציית אקטיבציה Softmax לצורך קבלת התפלגות ההסתברויות של המחלקות השונות בכל פיקסל.

יש לציין כי הרשת לומדת לבצע סגמנטציה לפי חלקים ולפי סוגים במקביל. כלומר, לרשת יש שני פלטים נפרדים עבור שתי הבעיות הנ"ל כך שניתן להשתמש באחד מהם גם עבור סגמנטציה בינארית. עקב כך, יש חיסכון משמעותי במספר המודלים השונים שיש לאמן.

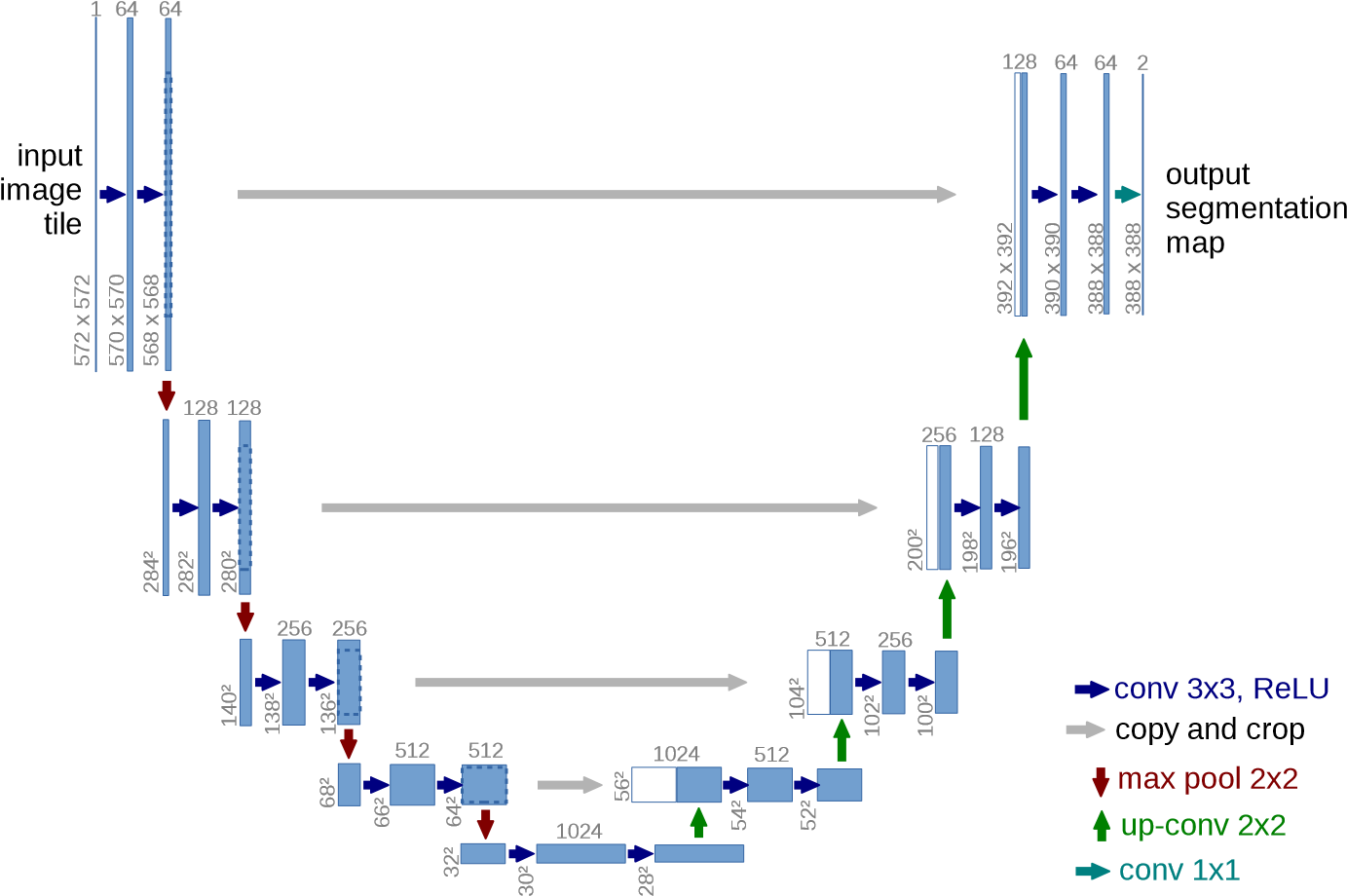
המודלים אומנו על תמונות ברזולוציה נמוכה, ונעשה שימוש בטכניקות של עיבוי נתונים: שינוי צבע, היפוך, תקריבים, סיבובים.



**איור 21** הטופולוגיה של הפתרון של NCT [1]. הקלט מתקבל בצד שמאל (תמונת RGB). בסוגריים ניתן לראות את גודל ה- kernel, מספר הערוצים, גודל ה- stride וגודל ה- padding, בהתאמה. ניתן לראות כי לרשת זו יש שני פלטים עבור שתי בעיות סגמנטציה – חלקים וסוגים. ניתן להשתמש באחד מהם גם עבור סגמנטציה בינארית. שני החיצים הארוכים באמצע מהווים את החיבורים השיוריים. חיבורים אלו משפרים את היכולת של הרשת לזהות את המיקום המדויק של האובייקטים בתמונה ובך משתפרת תוצאת הסגמנטציה.

הפתרון של UB:

פתרון זה מבוסס על ארכיטקטורת FCN מדורגת (cascaded). כלומר, הרשת מורכבת משלוש תת-רשתות כך שהרשת הראשונה לומדת לבצע סגמנטציה בינארית, והשנייה והשלישית מבצעות סגמנטציה לפי חלקים\סוגים ע"י שימוש בפלט של התת-רשת הראשונה, בשילוב עם התמונה המקורית, כך שיתאפשר מיסוך של הרקע. שיטה זו בעצם מסתמכת על העובדה שכדי לבצע סגמנטציה לפי חלקים\סוגים, יש צורך ללמוד קודם להפריד בין הרקע לבין המכשירים ומכאן שיש היגיון במבנה המדורג.



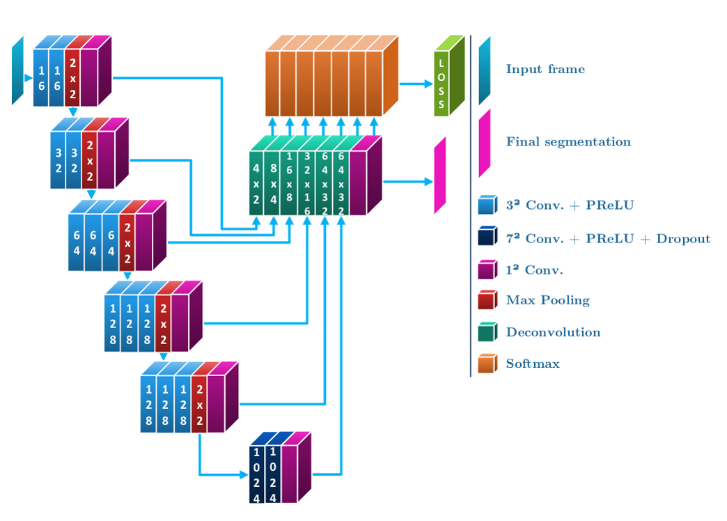
**איור 22** הטופולוגיה של ארכיטקטורת U-Net [6]. ארכיטקטורה זו מבצעת העתקה של מפות מאפיינים מהחלק המקודד אל החלק המפענח (ראה חיצים אפורים ארוכים) על מנת שהרשת תוכל ללמוד בצורה טובה יותר את המיקום המדויק של האובייקטים בתמונה המקורית כך שתוצאת הסגמנטציה תהיה מדויקת יותר [6]. ארכיטקטורה זו נחשבת לטובה במיוחד עבור בעיות סגמנטציה כאשר כמות הנתונים מוגבלת [2]. רשתות המבוססות על המבנה הכללי הנ"ל הוכיחו את תועלתן בביצוע סגמנטציה בינארית של מקבצי תמונות מתחומים שונים [3].

המבנה של כל אחת משלושת תת-הרשתות זהה ומבוסס על ארכיטקטורת מקודד-מפענחהדומה לארכיטקטורת U-Net [6] (ראה איור 22) כאשר השכבות המרכיבות אותן בנויות מבלוקים שיוריים אשר עושים שימוש בקונבולוציה, אקטיבציה ReLU ונרמול אצווה. לאחר כל בלוק בחלק המקודד יש שכבת Max pooling לצורך הקטנת הפלט, ולפני כל בלוק בחלק המפענח יש שכבת קונבולוציה משוחלפת לצורך הגדלת הקלט.

המודלים אומנו על תמונות ברזולוציה נמוכה, ונעשה שימוש בטכניקות של עיבוי נתונים: היפוך, הזזות, סיבובים, נרמול.

הפתרון של UCL:

הארכיטקטורה שהוצעה ע"י צוות זה נקראת ToolNet [9] (ראה איור 23). רשת זו מבצעת קונבולוציות משוחלפות עם גדלי stride שונים בהתאם לרזולוציה של מפות המאפיינים המתקבלות לאחר כל פעולת Max pooling. כמו כן, יש שכבת קונבולוציה שמקבלת כקלט את השרשור של כל מפות המאפיינים שהתקבלו לאחר הקונבולוציות המשוחלפות. כלומר, הרשת לומדת לבצע סגמנטציה ברזולוציות וברמות אבסטרקציה שונות במקביל, כאשר ההפסד הכולל מורכב מסכום ממושקל של ההפסדים המושרים ע"י ההתפלגויות המתקבלות משכבות ה- Softmax.

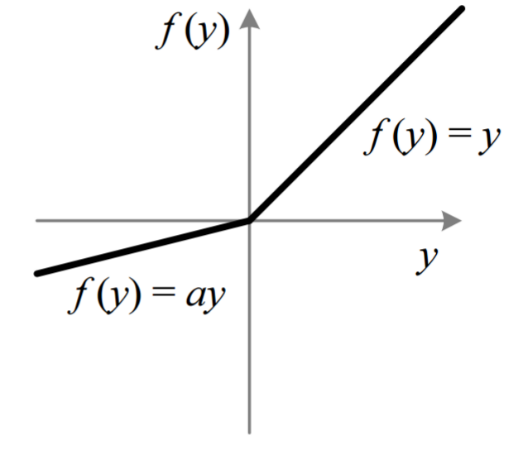


**איור 23** הטופולוגיה של הפתרון של UCL [1]. המספרים רשומים במאונך. בשכבות הקונבולוציה רשום מספר הערוצים. בשכבות הקונבולוציה המשוחלפת מספר הערוצים רשום מעל ה- x, וגודל ה- stride רשום מתחת ל- x.

פתרון זה מהיר יחסית עקב השימוש במספר קטן של ערוצים ברזולוציות הנמוכות תוך שימוש חוזר ונשנה בקונבולוציה המקטינה את מספר הערוצים לפני ביצוע הקונבולוציות המשוחלפות.

נעשה גם שימוש בפונקציית אקטיבציה PReLU (ראה איור 24) במקום ReLU רגילה לצורך שיפור הדיוק.

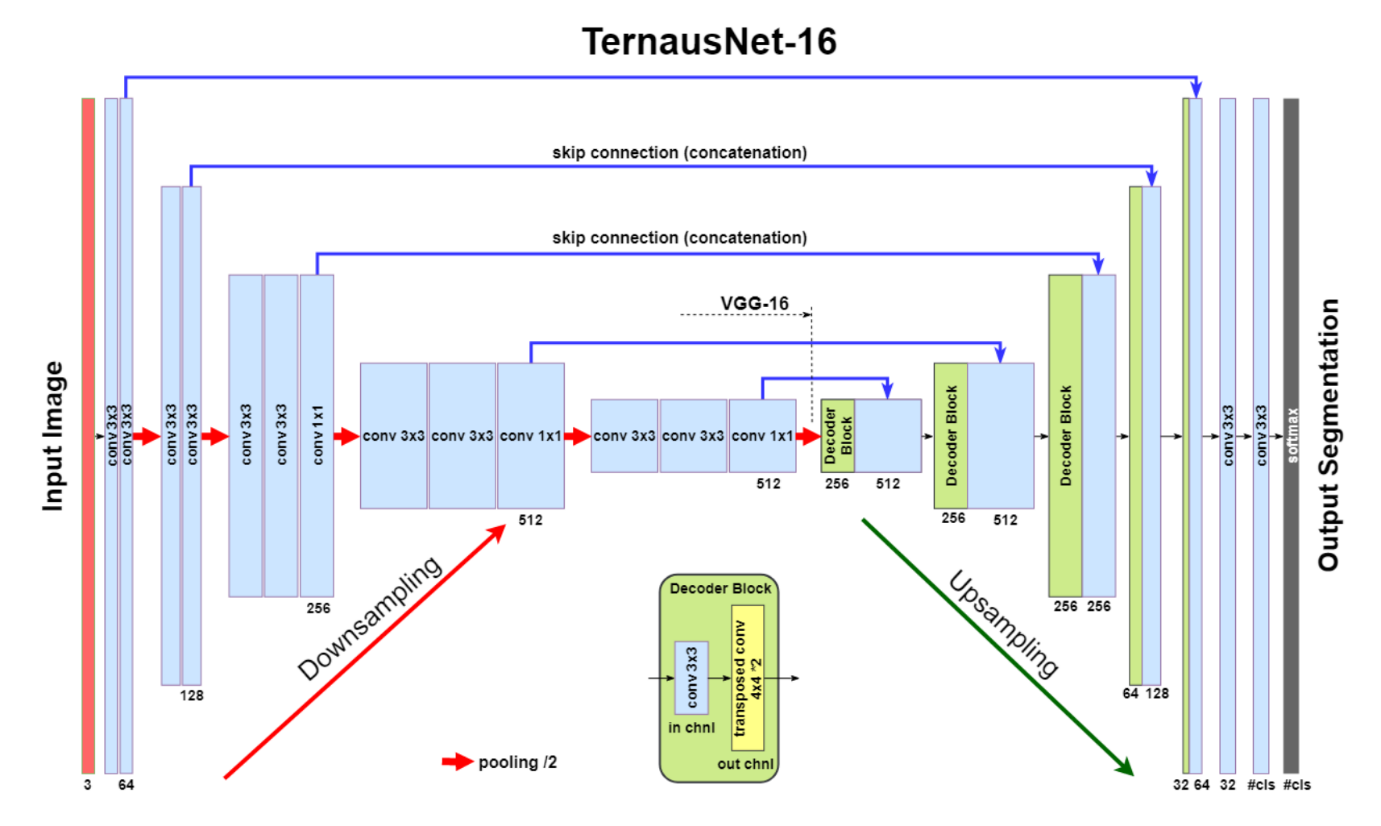
המודלים אומנו על תמונות ברזולוציה בינונית, ולא נעשה שימוש בטכניקות של עיבוי נתונים.



**איור 24** פונקציית אקטיבציה PReLU (Parametric ReLU). ניתן ללמוד את הפרמטר בכל שכבה, ובכך לשפר את הדיוק במחיר קטן של תקורה [9].

## **ארכיטקטורת** **TernausNet-16**

אציג כעת את הפתרון של MIT אשר ניצח בתחרות (ראה איור 25). פתרון זה מבוסס על ארכיטקטורת TernausNet [3] אשר מבוססת על ארכיטקטורת U-Net עם מקודד שאומן מראש על ImageNet. TernausNet-16 עושה שימוש במקודד VGG16.



**איור 25** הטופולוגיה של הפתרון המנצח של MIT [2]. לאחר פעולות הקונבולוציה יש אקטיבציה ReLU (חסר באיור). בדומה לארכיטקטורת U-Net, ישנם חיבורים שמעתיקים את מפות המאפיינים הצבועות בכחול מהחלק המקודד אל החלק המפענח. מפות אלו משורשרות עם המפות הצבועות בירוק אשר מתקבלות מהפלט של הבלוק המפענח (decoder block). #cls זה מספר המחלקות.

הרשת מקבלת כקלט תמונת RGB (הפס האדום באיור 25). לאחר מכן, יש שתי שכבות קונבולוציה עם kernel בגודל 3x3 ו- 64 ערוצים כל אחת (שני הפסים הכחולים הבאים), ומתבצעת פעולת Max pooling בגודל 2x2 עם stride בגודל 2 (חץ אדום קטן). באופן דומה, ממשיכים עם שתי שכבות עם 128 ערוצים, שלוש שכבות עם 256 ערוצים, (השכבה השלישית עם kernel בגודל 1x1), וכך הלאה (סה"כ 13 שכבות קונבולוציה) – עד כאן המקודד VGG16 (ניתן לראות היכן חלק זה מסתיים לפי הקו המקווקו באיור 25). לשם הפשטות, בחלק המפענח נעשה שימוש בבלוק מפענח (decoder block) אשר מורכב משתי שכבות – שכבת קונבולוציה עם kernel בגודל 3x3 ושכבת קונבולוציה משוחלפת עם kernel בגודל 4x4 ו- stride בגודל 2 – ניתן להסיק את מספר הערוצים של כל אחת משכבות אלו ("in chnl" ו- "out chnl") לפי המספרים הרשומים בשכבה הקודמת ובשכבת הבלוק, בהתאמה. למשל, בבלוק הראשון יש 512 ערוצים בשכבת הקונבולוציה ו- 256 ערוצים בשכבת הקונבולוציה המשוחלפת. משרשרים לבלוקים של המפענח את מפות המאפיינים בעלות הרזולוציה המתאימה מהחלק המקודד (בהתאם לחיצים הכחולים הארוכים). ניתן לראות כי מספר הערוצים בשכבות המפענח הולך וקטן (בהתאם למספרים הרשומים מתחת לשכבות) כך שבשכבה האחרונה הרזולוציה תהיה זהה לרזולוציית התמונה המקורית, ומספר הערוצים ישווה למספר המחלקות (בסיפא של הרשת יש שתי שכבות קונבולוציה עם kernel בגודל 3x3 – בראשונה יש 32 ערוצים ובאחרונה מספר הערוצים שווה למספר המחלקות). כרגיל, בשכבה האחרונה מופעלת פונקציית אקטיבציה Softmax לצורך קבלת הפרדיקציה עבור הפיקסלים.

בדומה לפתרונות קודמים, השימוש במקודד שאומן מראש מאפשר השגת תוצאות טובות למרות הגודל המזערי של קבוצות האימון בבעיה הנדונה. יתרון מרכזי נוסף של פתרון זה הוא השימוש בחיבורים שמעתיקים מפות מאפיינים מהחלק המקודד ומשרשרים אותן למפות המאפיינים בחלק המפענח כפי שנעשה בארכיטקטורת U-Net. כאמור, שיטה זו משפרת משמעותית את היכולת של הרשת לאחזר את המיקום המדויק של האובייקטים הרלוונטיים בתמונה ובכך מתקבל שיפור משמעותי בדיוק הסגמנטציה. למרבה הצער, לשיטה זו יש גם חיסרון מכיוון שמספר הערוצים הכולל בשכבות המפענח גדל משמעותית ומכאן שזמן הריצה של הרשת מתארך. באופן כללי, מספר הערוצים ברשת זו די גדול בשכבות המפענח. עקב האמור לעיל, הפתרון של MIT איטי יחסית.

חישוב ההפסד נעשה באמצעות שילוב של אינדקס ג'קארד (Jaccard index) ו*פונקציית אנטרופיה צולבת קטגורית (*categorical cross entropy*) [G].*

אינדקס ג'קארד מגדיר מדד של דמיון בין שתי קבוצות:

ניתן להכליל את מדד זה באמצעות מיסוך של ההסתברויות עבור כל מחלקה בנפרד [G]:

*כאשר ו- הם טנסורים בעלי הצורה של ההסתברויות החזויות ומסכה (*mask*) של התוויות הנכונות, בהתאמה ( זה גודל ה-* batch*, זה מספר המחלקות, ו- זה מספר הפיקסלים בתמונה) [G]. כלומר, מחשבת את מידת הדמיון בין הפלט לתוויות הנכונות ביחס למחלקה .*

***הערה****:* הפעולה זו מכפלת הדמר (Hadamard), וסימן הסכימה מחשב את סכום איברי הטנסור. כלומר, מתקיים כאשר זו ההסתברות שפיקסל מקלט של ה- batch שייך למחלקה , ו- שווה ל- 1 אם פיקסל זה אכן שייך למחלקה עפ"י התוויות הנכונות. אחרת, שווה ל- 0.  *מהווה קבוע חיובי קטן מאוד שתפקידו למנוע חלוקה ב- 0.*

*פונקציית אנטרופיה צולבת קטגורית לעומת זאת, מחשבת את ממוצע ההפסד לפיקסל:*

*לבסוף, הגדירו את פונקציית ההפסד [G]:*

*כאשר מהווה סכום של הפסדי אי-דמיון ברמת המחלקות, ו- זה היפר-פרמטר (*hyperparameter*) המאזן בין ל- [G].*

*לכן, מזעור של גורר מיקסום של הדמיון בין התוויות הנכונות לחזויות, ומיקסום של ההסתברויות הנכונות עבור כל פיקסל [2].*

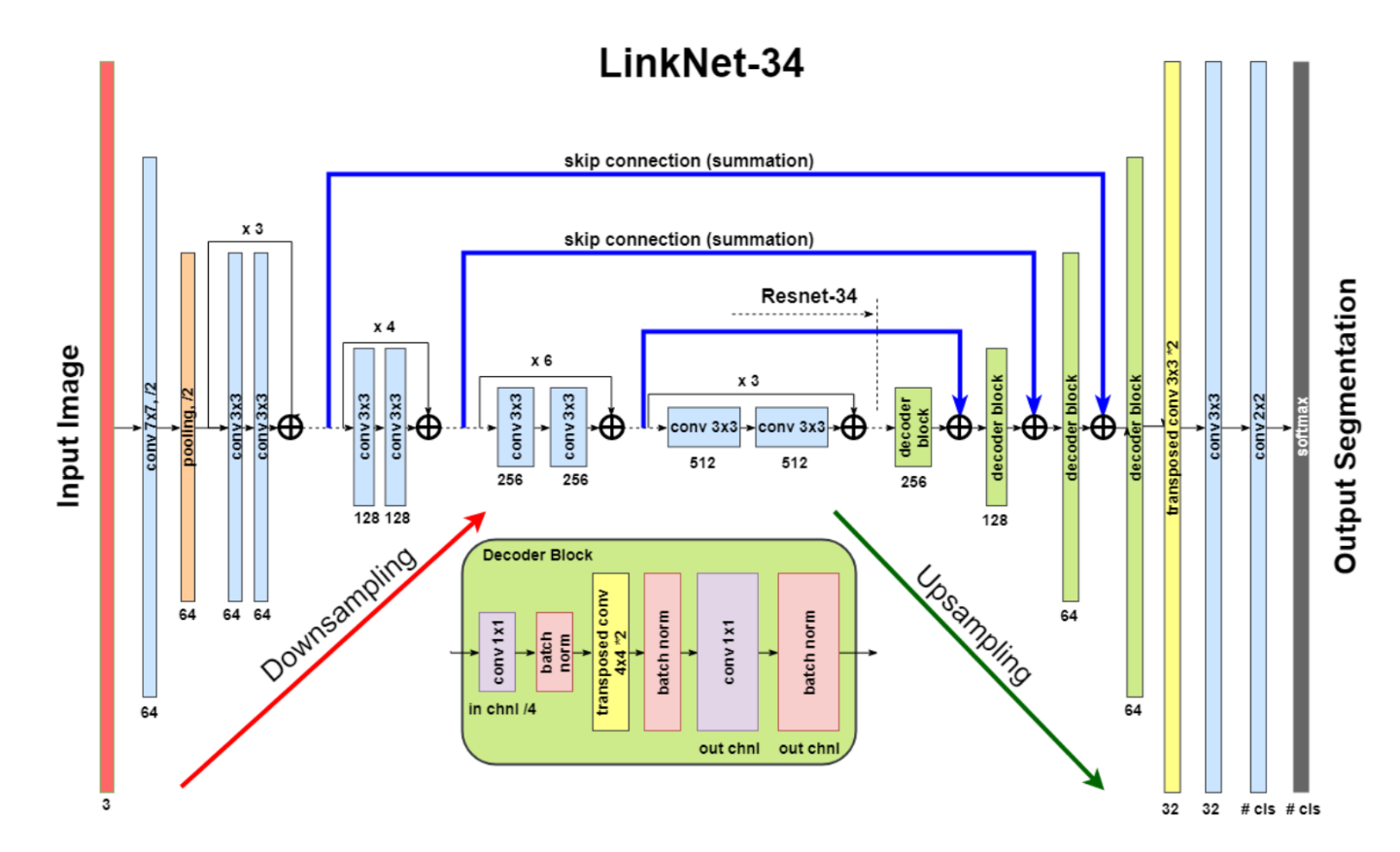
כדי להעריך את טיב המודל, ביצעו את האימות באמצעות 4-fold cross-validation כך שתמונות השייכות לאותו מקבץ יהיו באותה קבוצה (אימון\אימות) תוך שמירה על איזון בין המחלקות (מכשירים) בכל אחת מהקבוצות [H]. כפי שהסברתי בפרק הראשון, יש דמיון בין תמונות השייכות לאותו מקבץ ולכן רצוי שיהיו באותה קבוצה. עקב כך, לא ניתן להשתמש ביותר מ- 8 folds *(כמספר מקבצי האימון)* *–* *עובדה זו אינה מהווה בעיה מכיוון שזמן האימון של המודלים ארוך מאוד ולכן אין זה פרקטי להשתמש במספר* folds *גדול. כמו כן, יש הרבה אפשרויות לבחור את ה-* folds *במקרה של* 4-fold cross-validation *ולכן ניתן לבחור את האפשרות אשר מפיקה את האיזון הטוב ביותר בין המחלקות.*

בפתרון זה נעשה שימוש בתמונות ברזולוציה גבוהה כדי לשפר את הדיוק. כמו כן, נעשה שימוש בטכניקות של עיבוי נתונים: היפוך, נרמול, שינוי צבע.

עד כה הצגתי את הפתרונות הרלוונטיים לתחרות EndoVis 2017. בתת-פרק הבא, אציג ארכיטקטורה נוספת שתוכננה גם היא ע"י הצוות של MIT.

## **ארכיטקטורת LinkNet-34**

אציג כעת פתרון נוסף שתוכנן ע"י הצוות של MIT (ראה איור 26) המבוסס על ארכיטקטורת LinkNet [4]. ארכיטקטורה זו מבוססת על ארכיטקטורת מקודד-מפענח עם חיבורים שיוריים ארוכים ומקודד ResNet-34 [10] (ראה איור 27) שאומן מראש על ImageNet.

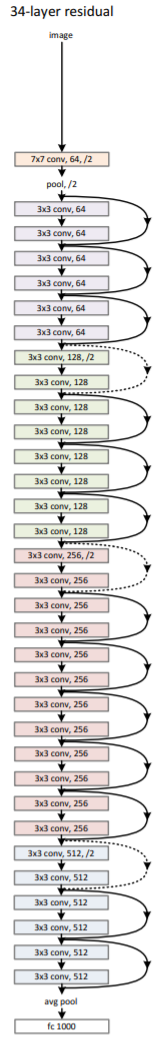


**איור 26** הטופולוגיה של ארכיטקטורת LinkNet-34 [2]. ניתן לראות את הבלוקים השיוריים של ResNet-34. המספר מעל כל בלוק שיורי מציין את מספר הבלוקים הרצופים. החיבורים בכחול מבצעים חיבור של מפות מאפיינים בדומה לבלוקים שיוריים. יש אקטיבציה ReLU אחרי כל שכבת נרמול אצווה או קונבולוציות רגילות\משוחלפות (חסר באיור).

בדומה לפתרון הקודם, הרשת מקבלת כקלט תמונת RGB ברזולוציה גבוהה. לאחר מכן, יש שכבת קונבולוציה עם kernel בגודל 7x7 עם 64 ערוצים ו- stride בגודל 2 לצורך הקטנת הרזולוציה. הרזולוציה מוקטנת שוב באמצעות Max pooling (הפס הכתום באיור 26). השכבות הכחולות הבאות מורכבות מארבע קבוצות של בלוקים שיוריים: קבוצה של 3, 4, 6, ו-3 בלוקים עם 64, 128, 256, ו- 512 ערוצים בכל שכבה (כל בלוק מורכב משתי שכבות קונבולוציה), בהתאמה, ו- kernel בגודל 3x3 כאשר בשכבה הראשונה בכל אחת מהקבוצות (מלבד זו הראשונה) הקונבולוציה מתבצעת עם stride בגודל 2 לצורך הקטנת הרזולוציה – עד כאן המקודד ResNet-34 (ראה קו מקווקו באיור 26). הבלוק המפענח מגדיר את השכבות הבאות (צבועות בירוק באיור 26): שכבת קונבולוציה עם kernel בגודל 1x1 כאשר מספר הערוצים שווה לרביעית ממספר הערוצים בשכבה הקודמת, נרמול אצווה, שכבת קונבולוציה משוחלפת עם אותו מספר של ערוצים, kernel בגודל 4x4 ו- stride בגודל 2, שוב נרמול אצווה, שכבת קונבולוציה עם kernel בגודל 1x1 עם מספר ערוצים בהתאם לרשום בשכבה המתאימה בחלק המפענח, ופעולת נרמול אצווה נוספת. המפות המהוות את הפלט של הבלוקים המפענחים מחוברות, מפה מול מפה, עם המפות בעלות הרזולוציה המתאימה בחלק המקודד (בהתאם לחיצים הכחולים הארוכים). אחרי הבלוקים, ישנה שכבת קונבולוציה משוחלפת עם kernel בגודל 3x3 ו- stride בגודל 2. השכבות בהמשך זהות לסיפא של TernausNet-16 (ה- kernel בגודל 2x2 בשכבה האחרונה).

באופן דומה לפתרון של NCT, החיבורים השיוריים הארוכים שעוברים מהמקודד אל המפענח משפרים את היכולת של הרשת ללמוד את המיקום המדויק של האובייקטים בתמונה. מצד אחד, התקורה הנוספת הכרוכה בחיבורים כאלו היא זניחה ביחס להעתקה של מפות מאפיינים מהמקודד אל המפענח (כפי שנעשה בארכיטקטורת U-net). מצד שני, העתקה של מפות משמרת את מלוא המידע ומכאן שחיבור המפות פחות מועיל מבחינת הדיוק. ניתן גם לראות כי מספר הערוצים בשכבות הראשונות והאחרונות הוא קטן יחסית ומרבית השכבות הן ברזולוציה נמוכה. כמו כן, נעשה שימוש בקונבולוציה בבלוק המפענח כדי להוריד משמעותית במספר הערוצים לפני ביצוע הקונבולוציה המשוחלפת. במילים פשוטות, ארכיטקטורה זו צרה ועמוקה ו- TernausNet-16 רחבה ורדודה יחסית. עקב האמור לעיל, זמן הריצה של הרשת מתקצר משמעותית ולכן יש לרשת זו יתרון משמעותי בהשוואה ל- TernuasNet-16.

חישוב ההפסד והערכת טיב המודל בוצעו באופן דומה לפתרון הקודם.



**איור 27**  הטופולוגיה של ארכיטקטורת ResNet-34 [10]. כל קשת מגדירה בלוק שיורי. אם הקשת מקווקות אז יש לקלט ולפלט של הבלוק מספר ערוצים ורזולוציה שונים. במקרים אלו ניתן להקטין את הרזולוציה של קלט הבלוק ולרפדו במפות מאופסות. לחלופין, ניתן להשתמש בקונבולוציה עם stride בגודל 2 כדי להתאים בין הממדים לפני ביצוע פעולת החיבור [10]. יש אקטיבציה ReLU לאחר כל פעולת קונבולוציה (חסר באיור).

# **תוצאות**

בפרק זה אציג את התוצאות שהתקבלו ע"י הצוות של MIT בשלב הערכת הביצועים של ארכיטקטורות TernausNet-16 ו- LinkNet-34, ולאחר מכן אציג חלק מהתוצאות שהושגו בתחרות EndoVis 2017 ע"י הפתרון המנצח בהשוואה לתוצאות של הפתרונות האחרים שהצגתי בפרק הקודם. ניתן לראות פירוט מלא של תוצאות התחרות ב- [1].

## **מדדי הערכת ביצועים**

את דיוק המודלים השונים בחנו באמצעות אינדקס ג'קארד\מדד (Intersection over Union). ניתן להציג את מדד זה באופן הבא:

כאשר (true positive) זה מספר הפיקסלים השייכים למחלקה שהרשת סיווגה למחלקה , (false positive) זה מספר הפיקסלים שאינם שייכים למחלקה שהרשת סיווגה למחלקה , ו- (false negative) זה מספר הפיקסלים השייכים למחלקה שהרשת סיווגה למחלקה אחרת.

בהינתן בעיית סגמנטציה ותמונה מסוימת, חישבו את ממוצע ה- מעל כל המחלקות הנבחרות הקיימות בתמונה. תמונות אשר אינן מכילות אף אחת מהמחלקות הנבחרות אינן נכללות בחישוב [1].

כזכור, לכל מקבץ נתוני אימון יש מקבץ מתאים של נתוני מבחן, ובנוסף יש שני מקבצי מבחן גדולים. ה- מחושב בנפרד עבור כל מקבץ נתונים ע"י חישוב ממוצע ה- מעל כל התמונות של המקבץ. לבסוף, חישבו את ממוצע ה- הכולל ע"י חישוב ממוצע ממושקל של המקבצים כאשר המשקלים נקבעים בהתאם לגדלי המקבצים השונים.

חלק מהצוותים עשו שימוש במדד מקובל נוסף בשלב הערכת טיב המודלים [2]. מדד זה נקרא מדד והוא מוגדר באופן דומה למדד :

## **ביצועי TernausNet-16 לעומת LinkNet-34**

הצוות של MIT הציג גם את זמן הריצה של הרשתות בנוסף למידת הדיוק (ראה טבלה 1).

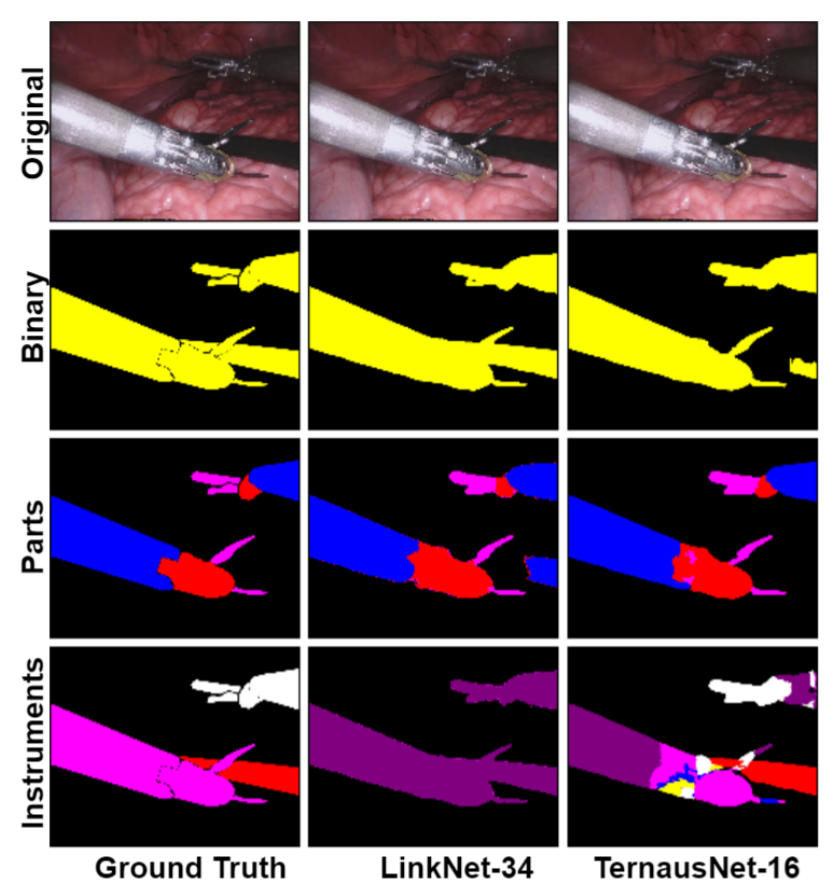
**טבלה 1** ביצועי המודלים של הפתרונות של MIT [2]. ניתן לראות את הדיוק וזמן הריצה עבור שלושת בעיות הסגמנטציה.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Segmentation problem | Binary | | | Parts | | | Type | | |
| Model | IOU | Dice | Time | IOU | Dice | Time | IOU | Dice | Time |
| TernausNet-16 | **83.60** | **90.01** | 184 | **65.50** | **75.97** | 202 | **33.78** | **44.95** | 275 |
| LinkNet-34 | 82.36 | 88.87 | **88** | 34.55 | 41.26 | **97** | 22.47 | 24.71 | **177** |

בטבלה 1 ניתן לראות את הביצועים של שני המודלים בכל אחת משלושת בעיות הסגמנטציה: בינארית (Binary), חלקים (Parts), וסוגים (Type). הדיוק נתון באחוזים וזמן הריצה נתון במילי-שניות (milliseconds). התוצאות הטובות ביותר מודגשות.

התוצאות שהושגו עבור בעיית הסגמנטציה הבינארית נראות די מבטיחות ואין הבדל משמעותי במידת הדיוק בין שתי הרשתות, אך נהיר כי זמן הריצה של LinkNet-34 קצר משמעותית מזה של TernausNet-16 [2]. עקב כך, יתכן כי ארכיטקטורת LinkNet-34 יותר פרקטית עבור בעיה זו. לעומת זאת, במקרה של בעיות הסגמנטציה האחרות, ארכיטקטורת TernuasNet-16 משיגה דיוק הרבה יותר גבוה. אי לכך, יתכן כי ארכיטקטורה זו סבירה יותר עבור בעיות אלו, אם כי תידרש חומרה מהירה במיוחד כדי לבצע את הסגמנטציה בזמן אמת. עדיין, מידת הדיוק אינה מרשימה במיוחד עקב הגודל המזערי של קבוצות האימון וחוסר ייצוג של מכשירים מסוימים בנתונים [2].

ניתן לראות באיור 28 כי שני הפתרונות מפיקים תוצאות סבירות עבור שתי הבעיות הראשונות (סגמנטציה בינארית וסגמנטציה לפי חלקים). מפתיע לראות כי LinkNet-34 מפיקה תוצאה מרשימה יותר מ- TernausNet-16 במקרה של סגמנטציה בינארית, אם כי מדובר בתמונה אחת בלבד. לעומת זאת, נהיר כי יש קושי רב בבעיית הסגמנטציה השלישית (סגמנטציה לפי סוגים). המודל של LinkNet-34 קבע מחלקה אחת עבור כל המכשירים, והמודל של TernausNet-16 קבע מספר מחלקות עבור כל מכשיר. יתכן בהחלט כי הארכיטקטורות הללו אינן מתוחכמות מספיק כדי להתמודד בצורה מספיק טובה עם בעיה זו.



**איור 28**  דוגמאות של תוצאות הסגמנטציה [2]. בעמודה השמאלית מופיעות התוויות הנכונות של בעיות הסגמנטציה השונות, ובשתי העמודות הימניות מופיעות התוצאות של שתי הרשתות.

יש לציין כי זמני הריצה הושגו באמצעות כרטיס מסך די חדיש [2] (NVIDIA GTX 1080Ti) ויש צורך בביצוע סגמנטציה של לפחות 25 תמונות בשנייה על מנת שהפתרון יהיה פרקטי בזמן אמת [9]. לאור שיקולים אלו, יתכן כי אפילו ארכיטקטורת LinkNet-34 אינה מהירה דיה במתכונתה הנוכחית. אולי כדאי להשתמש בתמונות ברזולוציה נמוכה יותר כדי להאיץ את הרשת.

## **תחרות EndoVis 2017**

אציג כעת את התוצאות שהושגו ע"י הצוותים בתחרות EndoVis 2017.

בכל טבלה מופיעים ערכי הדיוק (ערכים בין 0 ל- 1) שהתקבלו על 10 קבוצות המבחן לפי מדד (קבוצות 9 ו- 10 הן קבוצות המבחן הגדולות). בשורה האחרונה מופיע ממוצע ממושקל. בנוסף, מוצגות היסטוגרמות מתאימות שיצרתי בתוכנת Excel.

סגמנטציה בינארית:

**טבלה 2** ביצועי המודלים של הצוותים השונים בבעיית הסגמנטציה הבינארית [1].

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test set | NCT | UB | MIT | SIAT | UCL | UA |
| 1 | 0.784 | 0.807 | **0.854** | 0.625 | 0.631 | 0.413 |
| 2 | 0.788 | **0.806** | 0.794 | 0.669 | 0.645 | 0.463 |
| 3 | 0.926 | 0.914 | **0.949** | 0.897 | 0.895 | 0.703 |
| 4 | 0.934 | 0.925 | **0.949** | 0.907 | 0.883 | 0.751 |
| 5 | 0.701 | 0.740 | **0.862** | 0.604 | 0.719 | 0.375 |
| 6 | 0.876 | 0.890 | **0.922** | 0.843 | 0.852 | 0.667 |
| 7 | 0.846 | **0.930** | 0.856 | 0.832 | 0.710 | 0.362 |
| 8 | 0.881 | 0.904 | **0.937** | 0.513 | 0.517 | 0.797 |
| 9 | 0.789 | 0.855 | **0.865** | 0.839 | 0.808 | 0.539 |
| 10 | 0.899 | **0.917** | 0.905 | 0.899 | 0.869 | 0.689 |
| Mean | 0.843 | 0.875 | **0.888** | 0.803 | 0.785 | 0.591 |

**היסטוגרמה 1**  היסטוגרמה של ערכי טבלה 2 שיצרתי בתוכנת Excel. ניתן לראות את ערכי הדיוק (ערכים בין 0 ל- 1) של הפתרונות השונים על המקבצים השונים.

ניתן לראות בהיסטוגרמה 1 כי רוב הפתרונות השיגו ממוצע דיוק טוב יחסית (~0.85), מלבד הפתרון של UA (~0.6) – עובדה זו לא מפתיעה מכיוון שפתרונם מבוסס על רשת FCN פשוטה יחסית אשר מבצעת הגדלת רזולוציה בקפיצות גדולות ללא שימוש במפות מאפיינים הנמצאות בחלקים הרדודים יותר של הרשת, ומכאן שיש מעבר מצומצם של מידע לגבי המיקום המדויק של האובייקטים בתמונה. כמו כן, ניתן לראות כי כל הפתרונות התקשו קצת בהשגת תוצאות טובות במקבצים מסוימים (מקבצים 1, 2 ו- 5). משום מה, הפתרונות של SIAT ו- UCL התקשו במיוחד עם מקבץ 8 (~0.5). הפתרון של MIT מוביל באופן עקבי ברוב המקבצים, אך הביצועים של UB טובים כמעט באותה מידה. יש לציין כי שני הפתרונות הנ"ל מבוססים על ארכיטקטורת U-Net – עובדה זו מעידה על התועלת בשרשור של מפות מאפיינים מהמקודד אל המפענח.

סגמנטציה לפי חלקים:

**טבלה 3** ביצועי המודלים של הצוותים השונים בבעיית הסגמנטציה לפי חלקים [1].

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test set | NCT | UB | MIT | SIAT | UCL | UA |
| 1 | 0.723 | 0.715 | **0.737** | 0.591 | 0.611 | 0.485 |
| 2 | 0.705 | 0.725 | **0.792** | 0.632 | 0.606 | 0.559 |
| 3 | 0.809 | 0.779 | **0.825** | 0.753 | 0.692 | 0.640 |
| 4 | 0.845 | 0.737 | **0.902** | 0.792 | 0.630 | 0.692 |
| 5 | 0.607 | 0.565 | **0.695** | 0.509 | 0.541 | 0.473 |
| 6 | 0.731 | 0.763 | **0.802** | 0.677 | 0.668 | 0.608 |
| 7 | 0.729 | **0.747** | 0.655 | 0.604 | 0.523 | 0.438 |
| 8 | 0.644 | 0.721 | **0.737** | 0.496 | 0.441 | 0.604 |
| 9 | 0.561 | 0.597 | 0.650 | **0.655** | 0.600 | 0.551 |
| 10 | **0.788** | 0.767 | 0.762 | 0.751 | 0.713 | 0.637 |
| Mean | 0.699 | 0.700 | **0.737** | 0.667 | 0.623 | 0.578 |

**היסטוגרמה 2**  היסטוגרמה של ערכי טבלה 3 שיצרתי בתוכנת Excel. ניתן לראות את ערכי הדיוק (ערכים בין 0 ל- 1) של הפתרונות השונים על המקבצים השונים.

כצפוי, ניתן לראות בהיסטוגרמה 2 כי בעיה זו קשה יותר לפי הדיוק הממוצע (~0.65). הפתרון של UA מתקשה יותר גם כאן, אך הפרשי הדיוק בינו לבין שאר המתמודדים מצומצמים יותר. מקבצים מסוימים נראים מאתגרים יותר גם בבעיה זו. מעבר לכך, לא ניתן לראות אנומליות כלשהן. הפתרון של MIT מוביל גם כאן.

סגמנטציה לפי סוגים:

**טבלה 4** ביצועי המודלים של הצוותים השונים בבעיית הסגמנטציה לפי סוגים [1].

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test set | NCT | UB | MIT | SIAT | UCL | UA |
| 1 | 0.056 | 0.111 | **0.177** | 0.138 | 0.073 | 0.068 |
| 2 | 0.499 | 0.722 | **0.766** | 0.013 | 0.481 | 0.244 |
| 3 | **0.926** | 0.864 | 0.611 | 0.537 | 0.496 | 0.765 |
| 4 | 0.551 | 0.680 | **0.871** | 0.223 | 0.204 | 0.677 |
| 5 | 0.442 | 0.443 | **0.649** | 0.017 | 0.301 | 0.001 |
| 6 | 0.109 | 0.371 | **0.593** | 0.462 | 0.246 | 0.400 |
| 7 | 0.393 | **0.416** | 0.305 | 0.102 | 0.071 | 0.000 |
| 8 | 0.441 | 0.384 | **0.833** | 0.028 | 0.109 | 0.357 |
| 9 | 0.247 | 0.106 | **0.357** | 0.315 | 0.272 | 0.040 |
| 10 | 0.552 | 0.709 | 0.609 | 0.791 | 0.583 | **0.715** |
| Mean | 0.409 | 0.453 | **0.542** | 0.371 | 0.337 | 0.346 |

**היסטוגרמה 3**  היסטוגרמה של ערכי טבלה 4 שיצרתי בתוכנת Excel. ניתן לראות את ערכי הדיוק (ערכים בין 0 ל- 1) של הפתרונות השונים על המקבצים השונים.

בבעיה זו ניכר כי מתחיל להיות קושי רב עבור כל הפתרונות (ממוצע דיוק ~0.45). חלק מהפתרונות משיגים דיוק ממש אפסי על חלק מהמקבצים. ניתן אף לומר כי כל הפתרונות כשלו במקבץ הראשון (הדיוק נמוך מ- ~0.2). נהיר גם כי מקבץ 9 הרבה יותר קשה ממקבץ 10 בבעיה זו (כזכור, אלו המקבצים הגדולים). בחלק מהמקבצים (מקבצים 4, 5 ו- 8) הפתרון של MIT משיג דיוק טוב בהרבה משאר המתמודדים, אך פתרונות אחרים מובילים במקבצים אחרים (מקבצים 3 ו- 10). באופן כללי, יש חוסר עקביות משמעותי בביצועים ולכן קשה להשוות בין הפתרונות בבעיה זו. סביר להניח כי המודלים "סובלים" מהתאמת יתר.

לסיכום, קל לראות כי הצוות של MIT השיג את התוצאות הטובות ביותר ברוב המקבצים, וגם את הדיוק הממוצע הטוב ביותר בשלושת הבעיות. כמו כן, אין ספק שבעיית הסגמנטציה הבינארית קלה יחסית, בעיית הסגמנטציה לפי סוגים קשה במיוחד, ורמת הקושי של בעיית הסגמנטציה לפי חלקים היא בינונית באופן יחסי. בהינתן כמות גדולה יותר של נתוני אימון, יתכן כי יהיה אפשר להשיג תוצאות מספיק טובות בשתי הבעיות הקלות יותר באמצעות הארכיטקטורות המוצלחות יותר. לעומת זאת, בבעיית הסגמנטציה לפי סוגים אני מתרשם שבהחלט יש צורך בפתרונות מתוחכמים יותר.

# **סיכום, דיון, מסקנות**

בפרק הראשון הצגתי את בעיית הסגמנטציה של מכשירים כירורגיים והצבעתי על חשיבותה בתחום הניתוחים הזעיר פולשניים תוך מתן דוגמאות לשימושים אפשריים ושימת דגש על האתגרים והמאפיינים הייחודיים לבעיה זו. אחד האתגרים המרכזיים בבעיית הסגמנטציה בכלל ובבעיה הנדונה בפרט היא הכמות המוגבלת של הנתונים. לכן, יש בהחלט צורך באיסוף מקבצי נתונים גדולים ומגוונים יותר מאלו שנעשה בהם שימוש בתחרות EndoVis 2017 על מנת שיהיה ניתן לבחון באופן מעמיק ומועיל את הגישות השונות אשר הוצעו כפתרון לבעיה. עיבוי נתונים היא שיטה מוכרת וטובה להתמודדות עם מחסור בנתונים לצורך אימון של מערכות לומדות, ולכן אין זה מפתיע כי רוב הצוותים עשו שימוש בשיטה זו. עם זאת, הנתונים בבעיה הנדונה הם ייחודיים מאוד ולכן קשה לדעת אילו טרנספורמציות תהיינה מועילות, או לפחות בלתי מזיקות. סביר להניח כי מפאת סיבה זו ושיקולים הקשורים למגבלת הזמן, כל צוות ביצע תהליך עיבוי מצומצם ושמרני יחסית בהתאם לשיקול דעתו. למשל, רוב הצוותים עשו שימוש בטרנספורמציות מרחביות כגון היפוך וסיבוב, אך נמנעו מטרנספורמציות המשפיעות על התאורה. יש בכך היגיון מכיוון שהמצלמה יכולה להסתובב והתאורה הגלובלית בסצנת הניתוח די אחידה [1]. יתכן כי כדאי לחקור בעתיד שיטות של עיבוי נתונים המתאימות לבעיה הנדונה.

בפרק השני הצגתי את הגישות השונות של הצוותים שהתמודדו בתחרות EndoVis 2017 אשר התבססו על ארכיטקטורה של רשת עמוקה מסוג FCN שנותנת מענה לכל אחת מבעיות הסגמנטציה שהוגדרו בתחרות. ארכיטקטורת FCN במבנה של מקודד-מפענח זו ארכיטקטורה פשוטה, פופולרית ומועילה לפתרון בעיית הסגמנטציה ולכן אין זה מפתיע כי רוב הצוותים עשו שימוש בגרסאות שונות של ארכיטקטורה כללית זו. ההבדל העיקרי בין הארכיטקטורות של הצוותים השונים נעוץ באופן שבו הרשת מפעפעת מידע קדימה לגבי המיקום המדויק של האובייקטים השונים הקיימים בתמונה באמצעות פעולות upsampling שונות כדוגמת פעולת הקונבולוציה המשוחלפת. אסכם את הגישות השונות: בגישה של UA מבצעים חיבור של מפות מאפיינים ברזולוציות שונות בשכבות העמוקות של החלק המקודד, בגישה של SIAT משתמשים באינדקסים של פעולת ה- Max pooling בחלק המקודד לצורך ביצוע פעולת ה- upsampling בחלק המפענח, בגישה של NCT משתמשים בחיבורים שיוריים ארוכים העוברים בין השכבות הרדודות לשכבות העמוקות של הרשת, בגישה של UCL לומדים לבצע את הסגמנטציה ברזולוציות וברמות אבסטרקציה שונות במקביל, ובגישות של UB ו- MIT מבצעים העתקה של מפות מאפיינים מהחלק המקודד אל החלק המפענח. שתי הגישות האחרונות מבוססות על ארכיטקטורת U-Net, אך בפתרון של MIT נעשה גם שימוש במקודד שאומן מראש. שיטה זו נחשבת למועילה במיוחד עבור בעיות סגמנטציה עם מערך נתונים מוגבל, ולכן אין זה לגמרי מפתיע כי הפתרון של הצוות של MIT ניצח בתחרות. למרבה הצער, בתחרות לא הוצגו זמני הריצה של הפתרונות השונים. לכן, בחרתי להתמקד ולהשוות בין הארכיטקטורה המנצחת של הצוות של MIT לבין ארכיטקטורה אלטרנטיבית ומהירה יחסית שתוכנה גם היא ע"י צוות זה.

בפרק השלישי הצגתי ופירשתי את התוצאות שהושגו ע"י הצוות המנצח בשלב האימון והאימות באמצעות שתי ארכיטקטורות שונות, ואת התוצאות שהושגו ע"י הצוותים בתחרות. הדיוק של הפתרונות השונים די גבוה בבעיית הסגמנטציה הבינארית, ולכן יתכן כי ניתן להסתפק ברשת פשוטה ומהירה יחסית במקרה זה. כלומר, יתכן כי צוואר הבקבוק ביחס לבעיה זו נמצא דווקא בחומרה ולאו דווקא במגבלת הנתונים. לעומת זאת, במקרה של בעיות הסגמנטציה האחרות ניכר שיש קושי רב בהשגת תוצאות מספקות מבחינת הדיוק. הדבר נכון במיוחד במקרה של בעיית הסגמנטציה לפי סוגים עקב הקושי בהבדלה בין המכשירים השונים. הרי המכשירים די דומים אחד לשני בצבעם ובצורתם הכללית. אולי בבעיה זו כדאי לשקול שימוש בעיבוד נוסף ופשוט של הפלט המתקבל מהרשת לצורך שיפור התוצאה. למשל, אם המכשירים השונים בתמונה מופרדים ע"י הרקע אז אפשר להשתמש בתווית השכיחה של הפיקסלים המרכיבים אותם לצורך קביעת הסיווג הסופי. לחלופין ואף טוב מכך, אפשר להשתמש בערכי הפרדיקציה עצמם ולבחור בתווית עם סכום ההסתברויות הגדול ביותר – כך לתוויות סבירות יותר תהיה השפעה גדולה יותר על סיווג המכשיר.

יש לציין כי לטעויות הנובעות מכשלים במודל יתכנו השלכות חמורות על תוצאות הניתוח ולכן ביצועיו חייבים להיות טובים במיוחד, הן במידת הדיוק והן בזמן הריצה. לאור זאת, יש בהחלט סיבה להמשיך ולחקור את תחום זה בעתיד, ואכן כך נעשה. ניתן למצוא ב- [A] סדנאות רלוונטיות נוספות מהשנים האחרונות. למשל, בסדנת EndoVis 2018 הוצע אתגר שבו יש לבצע גם סגמנטציה של האיברים\רקמות הגוף בנוסף למכשירים – אין ספק שיש בכך צורך להבנה מלאה של סצנת הניתוח ע"י המחשב [1]. בסדנת EndoVis 2019 הוצע עוד אתגר בבעיית הסגמנטציה של מכשירים כירורגיים עם מערך נתונים גדול ומורכב יותר. לאחרונה הוצעה ארכיטקטורה חדישה ומתוחכמת שנקראת ISINet [15] המבוססת על ארכיטקטורה שנקראת Mask R-CNN [14]. ISINet משלבת איתור באמצעות תיבה תוחמת (bounding box) עם סגמנטציה מבוססת מופעים תוך שימוש במידע טמפורלי. ארכיטקטורה זו השיגה תוצאות מרשימות על מערך הנתונים של סדנת EndoVis 2017.

# **מקורות**

## **מאמרים**

[1] Max Allan, Alex Shvets, Thomas Kurmann, Zichen Zhang, Rahul Duggal, Yun-Hsuan Su, Nicola Rieke, Iro Laina, Niveditha Kalavakonda, Sebastian Bodenstedt, Luis Herrera, Wenqi Li, Vladimir Iglovikov, Huoling Luo, Jian Yang, Danail Stoyanov, Lena Maier-Hein, Stefanie Speidel, Mahdi Azizian, “2017 Robotic Instrument Segmentation Challenge”, <https://arxiv.org/abs/1902.06426>

[2] Alexey Shvets, Alexander Rakhlin, Alexandr A. Kalinin, Vladimir Iglovikov, “Automatic Instrument Segmentation in Robot-Assisted Surgery Using Deep Learning”, <https://arxiv.org/abs/1803.01207>

[3] Vladimir Iglovikov, Alexey Shvets, “TernausNet: U-Net with VGG11 Encoder Pre-Trained on ImageNet for Image Segmentation”, <https://arxiv.org/abs/1801.05746>

[4] Abhishek Chaurasia, Eugenio Culurciello, “LinkNet: Exploiting Encoder Representations for Efficient Semantic Segmentation”, <https://arxiv.org/abs/1707.03718>

[5] M. Allan, S. Ourselin, D. J. Hawkes, J. D. Kelly, D. Stoyanov, “3-D Pose Estimation of Articulated Instruments in Robotic Minimally Invasive Surgery”, <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8295119>

[6] Olaf Ronneberger, Philipp Fischer, Thomas Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation”, <https://arxiv.org/abs/1505.04597>

[7] Jonathan Long, Evan Shelhamer, Trevor Darrell, “Fully Convolutional Networks for Semantic Segmentation”, <https://arxiv.org/abs/1411.4038>

[8] Vijay Badrinarayanan, Alex Kendall, Roberto Cipolla, “SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation”, <https://arxiv.org/abs/1511.00561>

[9] Luis C. Garcia-Peraza-Herrera, Wenqi Li, Lucas Fidon, Caspar Gruijthuijsen, Alain Devreker, George Attilakos, Jan Deprest, Emmanuel Vander Poorten, Danail Stoyanov, Tom Vercauteren, Sebastien Ourselin, “ToolNet: Holistically-Nested Real-Time Segmentation of Robotic Surgical Tools”, <https://arxiv.org/abs/1706.08126>

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, “Deep Residual Learning for Image Recognition”, <https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html>

[11] Sergey Ioffe, Christian Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift”, <https://arxiv.org/abs/1502.03167>

[12] Vincent Dumoulin, Francesco Visin, “A guide to convolution arithmetic for deep learning”, <https://arxiv.org/abs/1603.07285>

[13] Hyeonwoo Noh, Seunghoon Hong, Bohyung Han, “Learning Deconvolution Network for Semantic Segmentation”, <https://arxiv.org/abs/1505.04366>

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick, “Mask R-CNN”, <https://arxiv.org/abs/1703.06870#:~:text=The%20method%2C%20called%20Mask%20R,CNN%2C%20running%20at%205%20fps>.

[15] C González, L Bravo-Sánchez, P Arbelaez, “ISINet: An Instance-Based Approach for Surgical Instrument Segmentation”, <https://arxiv.org/pdf/2007.05533.pdf>

## **מקורות נוספים**

[A] <https://endovis.grand-challenge.org/>

[B] <https://www.tensorflow.org/tutorials/images/transfer_learning>

[C] <https://neurohive.io/en/popular-networks/vgg16/>

[D] <http://image-net.org/index>

[E] <http://host.robots.ox.ac.uk/pascal/VOC/voc2012/>

[F] <https://towardsdatascience.com/dropout-on-convolutional-layers-is-weird-5c6ab14f19b2>

[G] <https://github.com/ternaus/robot-surgery-segmentation/blob/master/loss.py>

[H] <https://github.com/ternaus/robot-surgery-segmentation/blob/master/prepare_train_val.py>

[I] <https://en.wikipedia.org/wiki/Intuitive_Surgical#da_Vinci_Surgical_System>

[J] <https://en.wikipedia.org/wiki/Convolutional_neural_network>