



دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر

پروژه کارشناسی

سامانه قطعه‌بندی تصاویر پزشکی

نگارش

مهدی نیک نژاد

استاد راهنما

سرکار خانم دکتر مریم امیرمزلقانی

تیر ۱۴۰۲

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

به نام خدا

تاریخ: تیر ۱۴۰۲

## تعهدنامه اصالت اثر



اینجانب مهدی نیک نژاد متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظارت و راهنمایی اساتید دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مآخذ ذکر گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است. در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، نسخه‌برداری، ترجمه و اقتباس از این پایان‌نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مآخذ بلامانع است.

مهدی نیک نژاد

امضا

تقدیم بہ پدر و مادر عزیزم کہ در تمام سختی ها و دشواری های زندگی ہمواره کنارم بوده اند.

## سپاس‌گزاری

از اساتید دلسوز و محترم؛ سرکار خانم دکتر امیرمزلقانی و جناب آقای دکتر جوانمردی که با صبر و حوصله، از هیچ کمکی در این مسیر از من دریغ ننمودند و زحمت راهنمایی این پایان‌نامه را بر عهده گرفتند؛ کمال تشکر و قدردانی را دارم.

مدی نیک‌نژاد  
تیر ۱۴۰۲

## چکیده

پیشرفت‌های سریع در زمینه تصویربرداری پزشکی تحولات اساسی در پزشکی ایجاد کرده است. برای مثال تشخیص بیماری به کمک رایانه که در آن قطعه‌بندی تصاویر پزشکی<sup>۱</sup> نقش مهمی دارد، دقیق‌تر شده است. با اینکه شبکه‌های عصبی پیچشی<sup>۲</sup> در سالهای گذشته به عملکرد عالی دست یافته‌اند، اما به دلیل محلی بودن ذاتی عملیات پیچشی<sup>۳</sup>، نمی‌توانند اطلاعات معنایی سراسری و دوربرد<sup>۴</sup> را به خوبی یاد بگیرند پس با توجه به افزایش علاقه به استفاده از سازوکارهای خودتوجهی<sup>۵</sup> در بینایی رایانه<sup>۶</sup> و توانایی آنها برای غلبه بر این مشکل، معماری ترنس یونت (TransUNet) پیشنهاد شد که اولین چارچوب قطعه‌بندی تصاویر پزشکی با استفاده از مبدل بینایی<sup>۷</sup> به عنوان یک کدگذار<sup>۸</sup> در معماری یو-شکل می‌باشد.

ترنس یونت در مقایسه با معماری های مختلف به نتایج خوبی دست می‌یابد؛ به همین دلیل در این پروژه، ما از آن به عنوان مدل پایه که یک معماری ترکیبی پیچشی-مبدلی<sup>۹</sup> دارد، استفاده می‌کنیم. این معماری از هر دو ویژگی در بدست آوردن اطلاعات مکانی با وضوح بالا توسط شبکه‌های عصبی پیچشی و اطلاعات معنایی سراسری توسط مبدل‌ها استفاده می‌کند. همه پژوهش‌ها بر روی مجموعه دادگان Kvasir-SEG، CVC-ClinicDB و Ph2 انجام شده است. ما در ابتدا نتایج موجود در مقاله را بازتولید می‌کنیم و سپس با اعمال تغییراتی مناسب در جهت بهبود معماری گام برمی‌داریم و نتایج را بررسی می‌کنیم. برخی از این تغییرات موفق و برخی دیگر ناموفق بوده‌اند. در نهایت یک سامانه تحت وب با محوریت معماری جدید ایجاد کردیم.

کد پروژه در آدرس مقابل در دسترس است: <https://github.com/mnn59/BSc>

## واژه‌های کلیدی:

یادگیری عمیق، بینایی رایانه، قطعه‌بندی تصاویر پزشکی، UNet، سازوکار توجه، Transformer، کولونوسکوپی، درموسکوپی

<sup>1</sup>Medical Image Segmentation

<sup>2</sup>Convolutional Neural Network (CNN)

<sup>3</sup>Intrinsic locality of convolution operation

<sup>4</sup>Global and Long-range Context

<sup>5</sup>Self-attention Mechanism

<sup>6</sup>Computer Vision

<sup>7</sup>Vision Transformer

<sup>8</sup>Encoder

<sup>9</sup>Hybrid CNN-Transformer

# فهرست مطالب

صفحه

عنوان

۱	مقدمه	۱
۵	۱-۱ ساختار پایان نامه	۵
۶	۲ ادبیات مسأله و کارهای پیشین	۶
۷	۱-۲ مقدمه	۷
۸	۲-۲ شبکه پیچشی یو-شکل	۸
۹	۳-۲ مبدل بینایی	۹
۱۱	۴-۲ مجموعه دادگان	۱۱
۱۳	۵-۲ بررسی کارهای مشابه	۱۳
۱۳	۱-۵-۲ قطعه‌بندی تصاویر پزشکی مبتنی بر شبکه‌های عصبی پیچشی	۱۳
۱۴	۲-۵-۲ مبدل‌های بینایی	۱۴
۱۵	۳-۵-۲ قطعه‌بندی تصاویر پولیپ	۱۵
۱۷	۳ رویکرد پروژه	۱۷
۱۸	۱-۳ مقدمه	۱۸
۱۸	۲-۳ مدل پایه	۱۸
۲۰	۱-۲-۳ سازوکار توجه	۲۰
۲۱	۲-۲-۳ مبدل به عنوان کدگذار	۲۱
۲۲	۳-۲-۳ چارچوب کلی ترنس‌یونت	۲۲
۲۵	۳-۳ ایده‌های جدید	۲۵
۲۵	۱-۳-۳ تغییر در معماری کدگشا	۲۵
۲۸	۲-۳-۳ تغییر تابع زیان	۲۸
۲۸	۳-۳-۳ جایگزین کردن بهینه‌ساز	۲۸
۳۲	۴-۳ معیارها	۳۲
۳۴	۵-۳ نتایج	۳۴
۳۶	۴ پیاده‌سازی سامانه نهایی	۳۶
۳۹	۵ جمع‌بندی و نتیجه‌گیری	۳۹
۴۰	۱-۵ نتیجه‌گیری	۴۰
۴۰	۲-۵ کارهای آینده	۴۰
۴۱	کتاب‌نامه	۴۱

۴۵ ..... واژه‌نامه‌ی فارسی به انگلیسی

۴۹ ..... واژه‌نامه‌ی انگلیسی به فارسی



شکل	فهرست تصاویر	صفحه
۱-۲	معماری یونت . . . . .	۸
۲-۲	معماری وی-آی-تی (ViT) . . . . .	۱۰
۳-۲	نمونه مجموعه دادگان . . . . .	۱۲
۱-۳	معماری ترنس یونت . . . . .	۲۲
۲-۳	معماری ترنس گسکید . . . . .	۲۶
۳-۳	نتایج کیفی در مجموعه داده CVC-ClinicDB . . . . .	۳۰
۴-۳	نتایج کیفی در مجموعه داده Kvasir-SEG . . . . .	۳۰
۵-۳	نتایج کیفی در مجموعه داده Ph2 . . . . .	۳۱
۶-۳	معیار ارزیابی آی-اُ-یو . . . . .	۳۲
۷-۳	معیار ارزیابی دایس . . . . .	۳۳
۱-۴	تصویر سامانه نهایی قبل از بارگذاری تصویر . . . . .	۳۸
۲-۴	تصویر سامانه نهایی بعد از بارگذاری تصویر . . . . .	۳۸

صفحه	فهرست جداول	جدول
۳۴	نتایج استفاده از توابع زیان مختلف	۱-۳
۳۵	نتایج استفاده از بهینه‌سازهای مختلف	۲-۳
۳۵	نتایج مقایسه مدل پایه و جدید	۳-۳

# فصل اول

## مقدمه

قطعه‌بندی تصاویر پزشکی<sup>۱</sup> یکی از مراحل حیاتی در تشخیص‌های قبل از درمان، حین درمان و ارزیابی پس از درمان در بیماری‌های مختلف می‌باشد و می‌توان آنرا به عنوان یک مسأله پیش‌بینی در نظر گرفت که نقشه‌های قطعه‌بندی ضایعات یا اندام‌ها<sup>۲</sup> را ایجاد می‌کند. با توسعه و استفاده روزافزون از روش‌های تصویربرداری پزشکی (اشعه ایکس<sup>۳</sup>، سی‌تی<sup>۴</sup>، ام‌آر‌آی<sup>۵</sup>، پت<sup>۶</sup>، آندوسکوپی<sup>۷</sup> و بسیاری موارد دیگر) وجود ابزارهایی برای استخراج خودکار این اطلاعات اهمیت پیدا کرده است. امروزه با بهبود سخت‌افزار، روش‌های یادگیری عمیق برای این کارها امکان پذیرتر شده است و بیشتر روش‌های پرکاربرد مبتنی بر یادگیری عمیق می‌باشند [۱].

شبکه‌های عصبی پیچشی<sup>۸</sup> به طور گسترده برای پروژه‌های قطعه‌بندی تصاویر پزشکی استفاده شده‌اند. به طور خاص، یونت<sup>۹</sup> در میان انواع شبکه‌های مختلف به دلیل تولید نقشه‌های قطعه‌بندی با وضوح بالا، عملکرد قابل‌توجهی در قطعه‌بندی این دسته از تصاویر از خود نشان داده است [۱، ۲]. با توجه به معماری کارآمد کدگذار-کدگشا در یونت، چند نوع معماری با الهام از آن مانند یونت++<sup>۱۰</sup>، یونت<sup>۱۱</sup>+۳، دی-سی یونت<sup>۱۲</sup> عملکرد چشمگیری در قطعه‌بندی تصاویر پزشکی نشان داده‌اند و موفقیت فوق‌العاده‌ای در طیف وسیعی از کاربردهای پزشکی مانند قطعه‌بندی اجزای قلب از تصاویر ام‌آر‌آی، قطعه‌بندی اندام‌های بدن از تصاویر سی‌تی و قطعه‌بندی پولیپ<sup>۱۳</sup> از ویدئوهای کولونوسکوپی<sup>۱۴</sup> به دست آورده‌اند [۱].

علیرغم عملکرد رضایت‌بخش روش‌های مبتنی بر شبکه‌های عصبی پیچشی و قدرت بازنمایی‌شان، این معماری‌ها در یادگیری وابستگی‌های دوربرد<sup>۱۵</sup> بین پیکسل‌های تصویر دارای محدودیت‌هایی هستند. در واقع این شبکه‌های عصبی پیچشی به عملکرد عالی دست یافته‌اند، اما به دلیل محلی بودن ذاتی

<sup>1</sup>Medical Image Segmentation

<sup>2</sup>Segmentation maps of Lesions or Organs

<sup>3</sup>X-ray

<sup>4</sup>Computed Tomography (CT)

<sup>5</sup>Magnetic Resonance Imaging (MRI)

<sup>6</sup>Positron Emission Tomography (PET)

<sup>7</sup>Endoscopy

<sup>8</sup>Convolutional Neural Network (CNN)

<sup>9</sup>UNet

<sup>10</sup>UNet++

<sup>11</sup>UNet 3+

<sup>12</sup>DC-UNet

<sup>13</sup>Polyp

<sup>14</sup>Colonoscopy

<sup>15</sup>Long-range dependencies

عملیات پیچشی، نمی‌توانند اطلاعات معنایی سراسری<sup>۱۶</sup> و دوربرد را به خوبی یاد بگیرند [۱، ۳]. یعنی اگر تصاویر حاوی اطلاعات ساختاری با تغییرات زیادی در شکل و بافت باشند، این شبکه‌ها عملکرد ضعیفی خواهند داشت. برای غلبه بر این محدودیت، برخی از معماری‌ها از سازوکار توجه<sup>۱۷</sup> در معماری خود استفاده می‌کنند تا نقشه ویژگی<sup>۱۸</sup> را برای قطعه‌بندی بهتر تصاویر پزشکی بهبود بخشند. اگرچه این روش‌های مبتنی بر توجه بهبود داشته‌اند، اما همچنان از استخراج وابستگی‌های دوربرد ناکافی رنج می‌برند [۱].

پیشرفتهای اخیر در مبدل‌های بینایی<sup>۱۹</sup> باعث کاهش محدودیت‌های مربوط به وابستگی‌های دوربرد، به ویژه در قطعه‌بندی تصاویر پزشکی شده است. مبدل‌ها بر سازوکار توجه تکیه دارند و ابتدا برای پیش‌بینی دنباله به دنباله<sup>۲۰</sup> در پردازش زبان طبیعی معرفی شدند. مبدل‌ها از خودتوجهی<sup>۲۱</sup> برای یادگیری همبستگی<sup>۲۲</sup> بین تمام شناسه‌های<sup>۲۳</sup> ورودی استفاده می‌کنند که آنها را قادر می‌سازد وابستگی‌های دوربرد را دریافت کنند. به دنبال موفقیت مبدل‌ها در پردازش زبان طبیعی، مبدل‌های بینایی یک تصویر را به وصله‌های غیر همپوشان<sup>۲۴</sup> تقسیم می‌کنند و آنها را به همراه جاسازی‌های مکانی<sup>۲۵</sup> وارد واحد مبدل می‌کنند. یک نمونه از کاربرد مبدل‌های بینایی معماری ترنس‌یونت<sup>۲۶</sup> می‌باشد که استخراج اطلاعات معنایی سراسری و اطلاعات مکانی را بهبود می‌بخشد این معماری از یک مبدل پیچشی ترکیبی به عنوان کدگذار جهت استخراج وابستگی‌های دوربرد و یک نمونه‌افزای آبخاری به عنوان کدگشا برای دریافت روابط محلی بین پیکسل‌ها استفاده می‌کند [۴].

اخیراً، مبدل‌های بینایی سلسله‌مراتبی، مانند مبدل سوئین<sup>۲۷</sup> با توجه مبتنی بر پنجره<sup>۲۸</sup> و مبدل بینایی هرمی<sup>۲۹</sup> با توجه کاهشی مکانی<sup>۳۰</sup> برای کاهش هزینه‌های محاسباتی معرفی شده‌اند. این نوع مبدل‌های

<sup>16</sup>Global context

<sup>17</sup>Attention Mechanism

<sup>18</sup>Feature map

<sup>19</sup>Vision Transformer (ViT)

<sup>20</sup>Sequence-to-sequence

<sup>21</sup>Self-attention

<sup>22</sup>Correlation

<sup>23</sup>Token

<sup>24</sup>Non-overlapped patches

<sup>25</sup>Positional embedding

<sup>26</sup>TransUNet

<sup>27</sup>Swin Transformer

<sup>28</sup>Window-based attention

<sup>29</sup>Pyramid Vision Transformer (PVT)

<sup>30</sup>Spatial Reduction Attention (SRA)

بینایی برای کار قطعه‌بندی تصاویر پزشکی بسیار موثر هستند. با این حال، به طور کلی خودتوجهی مورد استفاده حتی در این مبدلها توانایی آنها را برای یادگیری روابط محلی بین پیکسلها محدود می‌کند. به همین خاطر معماری‌های جدیدی مثل سگ‌فورمر<sup>۳۱</sup>، یو‌فورمر<sup>۳۲</sup> و پی‌وی-تی-وی‌تو<sup>۳۳</sup> سعی کردند با جاسازی لایه‌های پیچشی در مبدلها بر این محدودیت غلبه کنند و این چنین، این معماری‌ها توانستند تا حدی روابط محلی بین پیکسلها را بیاموزند [۸].

با در نظر گرفتن این مسائل، ما برای بهبود مدل پایه یک کدگشای مبتنی بر توجه جدید به نام گسکید را معرفی می‌کنیم که از بازنمایی سلسله مراتبی مبدلهای بینایی استفاده می‌کند. این کدگشا از درگاه توجه<sup>۳۴</sup> و پودمان‌های توجه پیچشی<sup>۳۵</sup> جهت اصلاح نقشه‌های ویژگی استفاده می‌کند و هر دو رابطه معنایی و مکانی را بین پیکسلها می‌آموزد.

ما در این پروژه، مدل ترنس‌یونت را به عنوان مدل پایه در کار خود استفاده کردیم و برای بهبود آن از معماری کدگشای گسکید<sup>۳۶</sup> استفاده کردیم و قرار است، تحقیقاتی که در مقالات مربوط به این دو صورت پذیرفته است را با استفاده از مجموعه دادگان خود، بازتولید کنیم. خواهیم دید که استفاده از مبدلها به تنهایی نتایج رضایت بخشی به دست نمی‌دهد، اما با استفاده از ویژگی‌های جاسازی شده بدست آمده از مدل از پیش آموزش دیده<sup>۳۷</sup> رزنت پنجاه<sup>۳۸</sup> می‌توان به شبکه کمک کرد. از جمله تغییراتی که جهت بهبود مدل اعمال کردیم، اضافه کردن تابع زیان<sup>۳۹</sup> جدید بود که بهبودی حاصل نکرد اما استفاده از کدگشای گسکید و همچنین تغییر بهینه‌ساز<sup>۴۰</sup> در مدل باعث بهبود در خروجی شد، لذا در سامانه نهایی خود از این مدل بهبودیافته استفاده کردیم.

---

<sup>31</sup>SegFormer

<sup>32</sup>UFormer

<sup>33</sup>PVTv2

<sup>34</sup>Attention Gate (AG)

<sup>35</sup>Convolutional Attention Module (CAM)

<sup>36</sup>CASCADE

<sup>37</sup>Pre-trained

<sup>38</sup>ResNet-50

<sup>39</sup>Loss function

<sup>40</sup>Optimizer

## ۱-۱ ساختار پایان نامه

در این زیربخش قصد داریم که مروری بر ساختار کلی این پایان نامه داشته باشیم و در مورد مطالب هر فصل به طور خلاصه توضیحاتی ارائه دهیم.

همانطور که قبل تر اشاره شد، رویکرد اصلی پروژه استفاده از مبدل ها و به طور کلی استفاده از سازوکار توجه در پردازش تصاویر است. لذا مناسب است که در خصوص این موارد توضیحاتی را ارائه دهیم. همچنین کارهای مشابه صورت گرفته می تواند راهگشای خوبی برای نوآوری در این موضوع باشد. در فصل اول در مورد ساختار مبدل های بینایی و سازوکار توجه به کار رفته در آنها توضیحاتی آورده شده است و سپس در ادامه با مجموعه دادگان به کار رفته در این پروژه و کارهای مشابه صورت گرفته، آشنا خواهیم شد.

در فصل سوم به یکی از کارهای انجام شده خواهیم پرداخت و محوریت پروژه بر روی این مقاله خواهد بود. معماری مورد استفاده در مقاله مورد نظر، به عنوان مدل پایه در نظر گرفته شده است و در همین فصل تغییرات انجام گرفته بر روی مدل پایه و نتایج حاصل از اجرای آن تغییرات را بر اساس معیارهایی که در همین فصل شرح داده شده اند ذکر خواهیم کرد.

فصل چهارم مربوط به سامانه تحت وب می باشد. در این فصل در مورد سامانه پیاده سازی شده توضیحاتی ارائه خواهد شد که تصویر ورودی را دریافت کرده و به عنوان خروجی تصویر ماسک<sup>۴۱</sup> را باز می گرداند. در این فصل به نحوه عملکرد این سامانه اشاره خواهد شد.

در آخرین فصل به مرور فصل ها خواهیم پرداخت و بعد از نتیجه گیری مختصر، به کارهای پیشنهادی در آینده اشاره خواهیم کرد.

<sup>41</sup>Mask

## فصل دوم

### ادبیات مسأله و کارهای پیشین



## ۱-۲ مقدمه

در این فصل قصد داریم که با معماری شبکه پیچشی یو-شکل به نام یونت، مبدل‌های بینایی و کارهای پیشین در حوزه قطعه‌بندی تصاویر پزشکی با استفاده از مبدل‌ها آشنا شویم. مناسب است که قبل از ورود به بحث کارهای پیشین، مختصری در مورد مجموعه دادگان مورد استفاده در پروژه شرح دهیم.

این فصل چند زیربخش دارد. همانطور که قبل‌تر توضیح داده شد، یونت عملکرد قابل توجهی در قطعه‌بندی تصاویر پزشکی نشان داده است [۱]. معماری آن شامل دو مسیر کاهشی<sup>۱</sup> و گسترشی<sup>۲</sup> و همچنین اتصالات پرش<sup>۳</sup> بین این دو مسیر جهت حفظ اطلاعات مکانی (محلی) می‌باشد که این شبکه را قادر می‌سازد تا تشخیص دهد که چه چیزی و در کجای تصویر قرار دارد. مبدل‌ها نیز برای اولین بار در مقاله واسوانی و همکاران در سال ۲۰۱۷ برای ترجمه ماشینی<sup>۴</sup> پیشنهاد شدند [۵] و در بسیاری از پژوهش‌های حوزه پردازش زبان طبیعی بهترین موفقیت‌ها را کسب کردند. برای اینکه مبدل‌ها برای کارهای بینایی رایانه<sup>۵</sup> نیز قابل استفاده باشند، پژوهش‌های زیادی صورت گرفته است که مهم‌ترین آنها مربوط به مبدل‌های بینایی بوده است. در این معماری استفاده از توجه سراسری<sup>۶</sup>، به نتایج خوبی در قطعه‌بندی تصاویر بر روی مجموعه داده ایمیجنت-۲۱کا<sup>۷</sup> رسیده است [۵]. رویکرد مربوط به این پروژه که به طور مفصل به آن خواهیم پرداخت نیز بر اساس ترکیب همین ایده‌ها می‌باشد. لذا مناسب است که قبل از ورود به بحث‌های اصلی با ادبیات موضوع یعنی شبکه یونت و وی-آی-تی<sup>۸</sup> و ماهیت دقیق آنها بیشتر آشنا شویم و در زیربخش بعدی همین فصل قرار هست که به همین موضوعات بپردازیم.

زیربخش بعد، مربوط به مجموعه دادگان مورد استفاده در این پژوهش است. در این زیربخش به سه مجموعه داده مناسب و پرکاربرد در حوزه قطعه‌بندی تصاویر پزشکی اشاره خواهیم کرد و تا حد ممکن آنها را شرح خواهیم داد.

سپس در زیربخش انتهایی این فصل به کارهای پیشین انجام شده در این حوزه اشاره کرده و ایده اصلی آنها را بیان خواهیم کرد.

<sup>1</sup>Contracting Path

<sup>2</sup>Expanding Path

<sup>3</sup>Skip connection

<sup>4</sup>Neural Machine Translation (NMT)

<sup>5</sup>Computer Vision

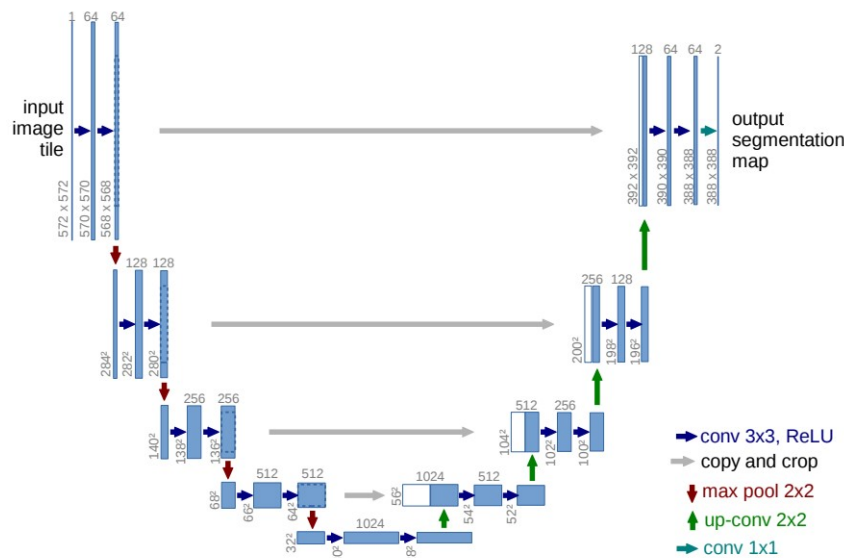
<sup>6</sup>Global Attention

<sup>7</sup>ImageNet-21k

<sup>8</sup>ViT

## ۲-۲ شبکه پیچشی یو-شکل

معماری شبکه یونِت اولین بار در مقاله [۶] با موضوع قطعه‌بندی تصاویر زیست پزشکی در سال ۲۰۱۵ توسط دانشگاه فریبورگ آلمان معرفی شد و توانست برگزیده چالش آی-اس-بی-آی ۲۰۱۵ شود. در شکل ۱-۲ معماری این شبکه نشان داده شده است و همانطور که مشخص هست از یک مسیر کاهشی (سمت چپ) و یک مسیر گسترشی (سمت راست) تشکیل شده است. شکل معماری حاصل شبیه به حرف یو<sup>۱۰</sup> در زبان انگلیسی می‌باشد و به همین دلیل نام آنرا یونِت گذاشته‌اند.



شکل ۱-۲: معماری یونِت [۶].

مسیر کاهشی از معماری معمولی یک شبکه پیچشی پیروی می‌کند و برای استخراج اطلاعات معنایی به کار می‌رود. این مسیر شامل استفاده مکرر از دو لایه پیچش ۳×۳ (پیچش‌های بدون لایه گذاری اضافی<sup>۱۱</sup> ) می‌باشد که در ادامه هر کدام یک واحد تابع فعال ساز رلو<sup>۱۲</sup> می‌آید و یک عملیات بیشترین ادغام<sup>۱۳</sup> ۲×۲ با گام<sup>۱۴</sup> دو برای نمونه کاهی<sup>۱۵</sup> انجام می‌شود. در هر مرحله از نمونه کاهی، تعداد کانالهای ویژگی<sup>۱۶</sup> دو برابر می‌شوند.

<sup>۹</sup>ISBI 2015

<sup>۱۰</sup>U

<sup>۱۱</sup>Unpadded Convolutions

<sup>۱۲</sup>Rectified linear activation function

<sup>۱۳</sup>Max pooling

<sup>۱۴</sup>Stride

<sup>۱۵</sup>Downsampling

<sup>۱۶</sup>Feature channels

مسیر گسترشی، محلی سازی<sup>۱۷</sup> دقیق را امکان پذیر می کند و به این واسطه اطلاعات مکانی استخراج می شوند. هر مرحله از این مسیر شامل سه مورد است: (۱) یک نمونه افزایی<sup>۱۸</sup> از نقشه ویژگی و به دنبال آن یک لایه پیچش  $2 \times 2$  که تعداد کانالهای ویژگی را به نصف کاهش می دهد. (۲) یک الحاق با نقشه ویژگی برش داده شده از مسیر کاهشی و (۳) دو لایه پیچش  $3 \times 3$  که در ادامه هر کدام یک رلو می آید. در لایه آخر یک لایه پیچش  $1 \times 1$  برای نگاشت هر بردار به تعداد مورد نظر کلاس استفاده می شود که اندازه ورودی تغییر پیدا نمی کند و فقط تعداد کانالها کاهش می یابد. در این پژوهش تعداد کلاسها برابر با دو می باشد؛ در واقع یکی مربوط به پس زمینه<sup>۱۹</sup> و دیگری مربوط به پیش زمینه<sup>۲۰</sup> می باشد [۶].

## ۳-۲ مبدل بینایی

در مقاله [۷] نشان داده شد که در حوزه بینایی رایانه، وابستگی به شبکه های عصبی پیچشی ضروری نیست و یک مبدل خالص که مستقیماً به دنباله ای از وصله های تصویر<sup>۲۱</sup> اعمال می شود نیز می تواند علی رغم داشتن منابع محاسباتی بسیار کمتر برای آموزش، در کار دسته بندی تصاویر به خوبی عمل کند. نمای کلی مدل در شکل ۲-۲ نشان داده شده است. به طور کلی یک تصویر را به وصله هایی<sup>۲۲</sup> با اندازه ثابت تقسیم می کنیم و هر کدام از وصله ها را جاسازی می کنیم و همچنین جاسازی های موقعیت<sup>۲۳</sup> را هم اضافه می کنیم و دنباله بردارهای حاصل را به کدگذاری استاندارد مبدل ها می دهیم. در طراحی مدل سعی شده است تا حد امکان از مبدل استاندارد پیروی شود.

<sup>17</sup>Localization

<sup>18</sup>Upsampling

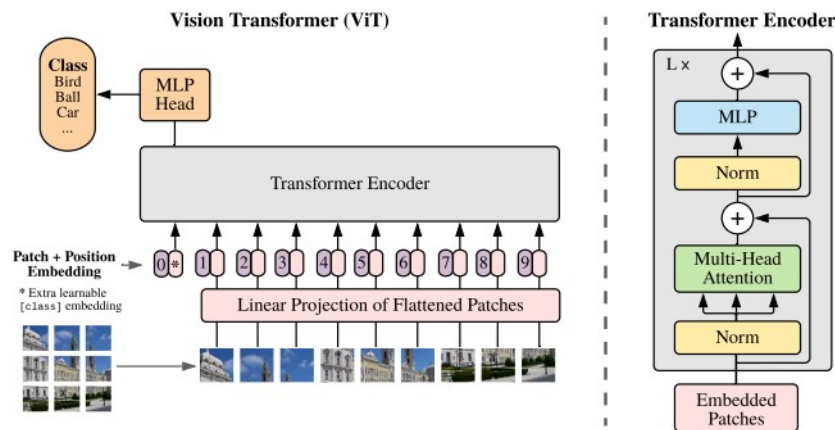
<sup>19</sup>Background

<sup>20</sup>Foreground

<sup>21</sup>Image Patches

<sup>22</sup>Patches

<sup>23</sup>Position embedding



شکل ۲-۲: معماری وی-آی-تی (ViT) [۷].

مبدل استاندارد، یک دنباله یک-بُعدی از جاسازی‌های شناسه<sup>۲۴</sup> را به عنوان ورودی دریافت می‌کند اما برای پردازش تصویر دو بعدی، آنرا به دنباله‌ای از وصله‌های دو بعدی مسطح<sup>۲۵</sup> تغییر شکل<sup>۲۶</sup> می‌دهد. در ادامه یک عملیات تبدیل خطی قابل یادگیری<sup>۲۷</sup> روی دنباله انجام می‌شود که عملاً یک ضرب ماتریسی است و باعث می‌شود طول بردار حاصل افزایش یابد. به خروجی این عملیات جاسازی وصله<sup>۲۸</sup> می‌گویند.

مشابه شناسه [سی-ال-اس]<sup>۲۹</sup> در مقاله پرت<sup>۳۰</sup>[۸]، یک شناسه قابل یادگیری به ابتدای دنباله‌ی وصله‌ها اضافه می‌شود و در ادامه، جاسازی‌های موقعیت به جاسازی‌های وصله نیز اضافه می‌شوند تا اطلاعات مکانی (موقعیتی) را حفظ کنند. دنباله حاصل از بردارهای جاسازی شده به عنوان ورودی به کدگذار مبدل داده می‌شود. کدگذار مبدل متشکل از لایه‌های خود توجهی چندسری<sup>۳۱</sup> و بلوک‌های پرسپترون چند لایه<sup>۳۲</sup> است. نرمال‌سازی لایه<sup>۳۳</sup> قبل از هر بلوک و اتصالات باقیمانده<sup>۳۴</sup> بعد از هر بلوک اعمال می‌شود. این عملیات  $L$  مرتبه تا رسیدن به خروجی انجام می‌شود و در نهایت برای انجام کار

<sup>۲۴</sup>Token embedding

<sup>۲۵</sup>Flattened

<sup>۲۶</sup>Reshape

<sup>۲۷</sup>Trainable Linear Projection

<sup>۲۸</sup>Patch Embedding

<sup>۲۹</sup>[CLS]

<sup>۳۰</sup>BERT

<sup>۳۱</sup>Multi-head self attention (MSA)

<sup>۳۲</sup>Multi Layer Perceptron (MLP)

<sup>۳۳</sup>Layer Normalization

<sup>۳۴</sup>Residual Connection

دسته‌بندی تصویر ورودی، به اولین وصله توجه می‌شود و بردار خروجی آن دریافت می‌شود [۷].

## ۴-۲ مجموعه دادگان

در این زیربخش قصد داریم که با مجموعه دادگانی که در این پروژه استفاده شده است، به صورت مختصر آشنا شویم. مجموعه دادگان مذکور، همگی دارای برچسب (ماسک) هستند و ما برای آموزش مدلها همگی را به نسبت هشتاد-بیست، دوبخشی<sup>۳۵</sup> کردیم. ماسک، یک تصویر دودویی<sup>۳۶</sup> است که نشان می‌دهد کدام پیکسل‌ها در تصویر اصلی متعلق به پولیپ یا ضایعه هستند و کدام‌ها نیستند. به طور خلاصه، برچسب‌های مجموعه داده ماسک‌های دودویی هستند که با ناحیه تحت پوشش پولیپ یا ضایعه مطابقت دارند.

مجموعه داده CVC-ClinicDB از فریم‌های استخراج شده از بین ۲۹ ویدئوی کولونوسکوپی جمع‌آوری شده توسط مرکز درمانی بیمارستان مرکزی بارسلونا در اسپانیا تشکیل شده است. این فریم‌ها حاوی چندین نمونه از پولیپ‌ها هستند. این مجموعه داده در چالش فرعی میکای ۲۰۱۵<sup>۳۷</sup> در موضوع تشخیص خودکار پولیپ در ویدیوهای کولونوسکوپی استفاده شده است و شامل ۶۱۲ تصویر با ابعاد ۳۸۴×۲۸۸ است [۹].

مجموعه داده Kvasir-SEG، شامل تصاویر پولیپ دستگاه گوارش و ماسک‌های قطعه‌بندی مربوطه است که توسط یک متخصص گوارش با تجربه به صورت دستی تأیید شده است. پولیپ‌ها پیش‌ساز سرطان کولورکتال<sup>۳۸</sup> هستند و تقریباً در نیمی از افراد در سن پنجاه سالگی که کولونوسکوپی غربالگری انجام می‌دهند، یافت می‌شود. نشان داده شده است که تشخیص زودهنگام پولیپ خطر ابتلا به سرطان کولورکتال را کاهش می‌دهد. بنابراین، تشخیص خودکار پولیپ‌های بیشتر در مراحل اولیه می‌تواند نقش مهمی در بهبود، پیشگیری و درمان سرطان کولورکتال داشته باشد. این انگیزه اصلی توسعه مجموعه داده Kvasir-SEG بوده است. این مجموعه داده شامل هزار تصویر به همراه ماسک‌های مربوطه می‌باشد و اندازه تصاویر آن از ۳۳۲×۴۸۷ تا ۱۹۲۰×۱۰۷۲ پیکسل متغیر است [۱۰].

<sup>35</sup> Split

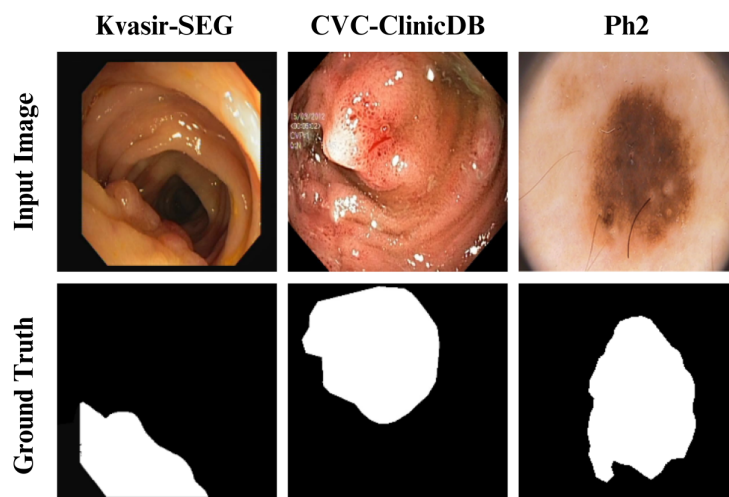
<sup>36</sup> Binary Image

<sup>37</sup> MICCAI 2015

<sup>38</sup> Colorectal cancer

افزایش ابتلا به سرطان پوست و بروز ملانوم<sup>۳۹</sup> (نوعی ضایعه پوستی)، توسعه سامانه‌های تشخیصی به کمک رایانه<sup>۴۰</sup> را برای دسته‌بندی تصاویر درموسکوپی ترویج کرده است. مجموعه داده Ph2 برای اهداف تحقیقاتی به منظور تسهیل مطالعات مقایسه‌ای بر روی الگوریتم‌های قطعه‌بندی و دسته‌بندی تصاویر درموسکوپی، توسعه یافته است. این مجموعه داده در بخش پوست‌شناسی<sup>۴۱</sup> بیمارستان پدرو هیسپانوی کشور پرتغال به دست آمده است. تصاویر درموسکوپی تحت شرایط یکسان از طریق سیستم آنالایزور مول تیوبینگر<sup>۴۲</sup> با استفاده از بزرگنمایی بیست برابر به دست آمده‌اند. آنها تصاویر رنگی با وضوح  $768 \times 560$  پیکسل هستند. این مجموعه داده شامل دویست تصویر از ضایعات ملانوسیتی است که شامل هشتاد خال معمول، هشتاد خال غیر معمول و چهل ملانوم می‌باشد[۱۱].

در تصویر شکل ۲-۳، سه نمونه از مجموعه داده‌گان را به همراه ماسک‌های مربوطه مشاهده می‌کنیم. ستون اول مربوط به مجموعه داده Kvasir-SEG، ستون دوم مربوط به مجموعه داده CVC-ClinicDB و ستون سوم مربوط به مجموعه داده Ph2 می‌باشد.



شکل ۲-۳: نمونه مجموعه داده‌گان

<sup>39</sup>Melanoma

<sup>40</sup>Computer-Aided Diagnosis (CAD)

<sup>41</sup>Dermatology

<sup>42</sup>Tuebinger Mole Analyzer

## ۵-۲ بررسی کارهای مشابه

در این زیربخش، قصد داریم که به کارهای مشابه انجام شده در این حوزه اشاره مختصری داشته باشیم. این کار می‌تواند باعث آشنایی با نوآوری‌های انجام‌شده در حل این نوع از مسائل شود. ابتدا متداول‌ترین روش‌های مبتنی بر شبکه‌های عصبی پیچشی مورد استفاده در قطعه‌بندی تصاویر پزشکی را خلاصه می‌کنیم، سپس کاربرد مبدل‌های بینایی را در سالهای اخیر در زمینه قطعه‌بندی تصاویر مرور می‌کنیم و در آخر به توضیح روشهای سنتی و مبتنی بر یادگیری عمیق در حوزه قطعه‌بندی تصاویر پولیپ می‌پردازیم.

## ۱-۵-۲ قطعه‌بندی تصاویر پزشکی مبتنی بر شبکه‌های عصبی پیچشی

شبکه‌های عصبی پیچشی، به‌ویژه شبکه یونیت با ساختار کدگذار-کدگشا و نسخه‌های مختلف آن، عملکرد فوق‌العاده‌ای را در قطعه‌بندی تصاویر پزشکی نشان داده‌اند و همچنین به دلیل سادگی و عملکرد خوب ساختار یو-شکل، انواع مختلفی از روشها مشابه یونیت دایما در حال ظهور هستند [۳]؛ برای مثال یونیت++ مجموعه‌ای از اتصالات پرش تودرتو و متراکم<sup>۴۳</sup> را معرفی می‌کند تا شکاف معنایی<sup>۴۴</sup> بین نقشه‌های ویژگی کدگذار و کدگشا قبل از الحاق<sup>۴۵</sup> پر شود [۱۲].

آتنشن-یونیت<sup>۴۶</sup> یک نوآوری جدید به نام درگاه توجه پیشنهاد می‌کند که مدل را قادر می‌سازد تا روی اهداف با اشکال و اندازه‌های مختلف تمرکز کند [۱۳]. در آخر یوتونیت<sup>۴۷</sup> که از بلوکهای باقیمانده یو-شکل<sup>۴۸</sup> استفاده می‌کند و دارای ساختار دو سطحی تودرتو می‌باشد به طوریکه در هر سطح آن از معماری یونیت استفاده شده است. این معماری قادر است اطلاعات معنایی بیشتری را دریافت کند؛ چراکه از ترکیبی از میدانهای تأثیر<sup>۴۹</sup> در اندازه‌های متفاوت استفاده می‌کند [۱۴].

<sup>43</sup> Nested and dense skip pathways

<sup>44</sup> Semantic gap

<sup>45</sup> Concatenation

<sup>46</sup> Attention UNet

<sup>47</sup> U2Net

<sup>48</sup> Residual U-Blocks (RSU)

<sup>49</sup> Receptive field

## ۲-۵-۲ مبدل‌های بینایی

با الهام از موفقیت مبدل‌ها در پژوهشهای مختلف در حوزه پردازش زبان طبیعی<sup>۵۰</sup>، روش‌های مبتنی بر مبدل‌ها بیشتر و بیشتر در پژوهشهای بینایی رایانه ظاهر شده‌اند. در میان مبدل‌های بینایی اخیر، وی-آی-تی اولین تلاشی است که ثابت می‌کند معماری خالص مبتنی بر مبدل‌ها می‌تواند بهترین روش موجود در حال حاضر<sup>۵۱</sup> در مورد پژوهش بازشناسی تصاویر<sup>۵۲</sup> از پیش آموزش دیده روی مجموعه دادگان بزرگ مانند ایمیجنت-۲۰۰۰<sup>۵۳</sup> و جی-اف-تی سیصد ام<sup>۵۴</sup> باشد [۱۵].

دیت<sup>۵۵</sup> [۱۶] استراتژی معلم-دانش‌آموز<sup>۵۶</sup> را مخصوص مبدل‌ها معرفی کرده است که بر اساس یک شناسه دیستیلیشن<sup>۵۷</sup> و فرآیند انتقال دانش تضمین می‌کند دانش‌آموز از طریق سازوکار توجه از مدل معلم آموزش ببیند. این پژوهش نشان می‌دهد که معماری دیت در مقایسه با معماری وی-آی-تی توانسته است به خوبی عمل کند و برای ایفای بهترین عملکرد خود در تولید یک مدل دسته‌بندی تصاویر، نیاز به حجم داده‌ها و منابع محاسباتی بسیار کمتری دارد؛ بطوریکه این معماری بر روی مجموعه دادگان کوچکتر ایمیجنت-۱۰۰۰<sup>۵۸</sup> آموزش دیده است و عملکرد خوبی داشته است [۱۵، ۱۶].

مبدل سوئین<sup>۵۹</sup> [۱۷] از سازوکار توجه پنجره لغزان<sup>۶۰</sup> استفاده می‌کند و نشان داده است با این کار می‌تواند مشکل پیچیدگی محاسباتی نمایی درجه دو<sup>۶۱</sup> در مبدل‌های بینایی را حل کند. این معماری با پیچیدگی محاسباتی خطی از طریق سازوکار خود توجهی مبتنی بر پنجره لغزان عملکرد خوبی را در بازشناسی تصویر، تشخیص اشیا<sup>۶۲</sup> و قطعه‌بندی معنایی<sup>۶۳</sup> به دست آورده است [۱۵، ۱].

یکی از مشکلات وی-آی-تی، ثابت ماندن و در واقع چند مقیاسی نبودن نقشه ویژگی‌ها است که باعث

<sup>50</sup>Natural Language Processing (NLP)

<sup>51</sup>State-of-the-arts (SOTA)

<sup>52</sup>Image Recognition

<sup>53</sup>ImageNet-22K

<sup>54</sup>JFT-300M

<sup>55</sup>DeiT (Data-efficient Image Transformer)

<sup>56</sup>Teacher-Student strategy

<sup>57</sup>Distillation token

<sup>58</sup>ImageNet-1K

<sup>59</sup>Swin Transformer

<sup>60</sup>Sliding window attention mechanism

<sup>61</sup>Quadratic complexity

<sup>62</sup>Object detection

<sup>63</sup>Semantic segmentation



می‌شود اطلاعات مکانی<sup>۶۴</sup> زیادی را از دست دهیم. برای بهبود این مشکل معماری پی-وی-تی<sup>۶۵</sup> [۱۸] یا مبدل بینایی هرمی معرفی شد که برای اولین بار، حالت چند مقیاسی را وارد معماری مبدل بینایی کرد. این معماری به معماری شبکه‌های عصبی پیچشی شباهت دارد و تنها تفاوتش در داشتن توجه سراسری<sup>۶۶</sup> در تمام مراحل است. بهبود مهم دیگری که پی-وی-تی نسب به وی-آی-تی دارد، معرفی یک سازوکار توجه جدید به نام توجه کاهشی مکانی<sup>۶۷</sup> است که توسط آن پیچیدگی محاسباتی بر مبنای ضریب کاهش<sup>۶۸</sup> تعریف شده تا حدی کاهش می‌یابد. اما چند وقت بعد نسخه دوم معماری مبدل بینایی هرمی با نام پی-وی-تی-وی-تو<sup>۶۹</sup> [۱۹] معرفی شد که سعی کرد عملکرد بهتر داشته باشد. در این نسخه، عملکرد نسخه قبل را با ترکیب یک لایه توجه با پیچیدگی خطی<sup>۷۰</sup>، جاسازی وصله هم‌پوشان<sup>۷۱</sup>، و یک شبکه پیچشی پیشخور<sup>۷۲</sup> بهبود بخشیدند [۱].

## ۳-۵-۲ قطعه‌بندی تصاویر پولیپ

### روشهای سنتی

تشخیص و بررسی دقیق زخم‌ها، پولیپ‌ها و تومورها در تصویربرداری آندوسکوپی به کمک رایانه جایگزین مؤثری برای تشخیص دستی است. راه‌حل‌های اولیه برای قطعه‌بندی پولیپ‌ها عمدتاً بر اساس ویژگی‌هایی مثل بافت یا ویژگی‌های هندسی بوده است؛ با این حال، این روش‌ها به دلیل شباهت زیاد پولیپ‌ها به بافت‌های اطراف، خطر تشخیص اشتباه بالایی دارند؛ بنابراین فنون یادگیری عمیق، توسعه پژوهشهای حوزه قطعه‌بندی پولیپ را تا حد زیادی ارتقا داده‌اند [۲۰].

### روشهای مبتنی بر یادگیری عمیق

از جمله این روشها می‌توان به معماری یونِت و یونِت++ اشاره کرد که پیش از این به آنها اشاره شد. معماری پُرانت<sup>۷۳</sup> [۲۱] یکی دیگر از روشهای تخصصی در حوزه قطعه‌بندی تصاویر پولیپ می‌باشد که از پودمان‌های

<sup>64</sup>Spatial information

<sup>65</sup>Pyramid Vision Transformer (PVT)

<sup>66</sup>Global attention

<sup>67</sup>Reduction factor

<sup>68</sup>PVTv2

<sup>69</sup>Linear complexity attention layer

<sup>70</sup>Overlapping Patch embedding

<sup>71</sup>Feedforward convolutional network

<sup>72</sup>PraNet

توجه معکوس<sup>۷۳</sup> برای استخراج اطلاعات مرزی از نقشه ویژگی‌های سراسری<sup>۷۴</sup>، که توسط یک کدگذاری جزئی موازی<sup>۷۵</sup> از ویژگی‌های سطح بالا تولید می‌شود، استفاده می‌کند. پولیپ-پی-وی-تی<sup>۷۶</sup>، معماری دیگری است که از مبدل بینایی هرمی به عنوان کدگذار استفاده می‌کند و سعی کرده است با تعریف سه پودمان با نامهای سی-اف-ام<sup>۷۸۷۷</sup>، سی-آی-ام<sup>۸۰۷۹</sup> و اس-ای-ام<sup>۸۲۸۱</sup>، به نتایج خوبی برسد. سی-اف-ام برای جمع‌آوری اطلاعات معنایی و مکانی پولیپ‌ها از ویژگی‌های سطح بالا، سی-آی-ام برای جمع‌آوری اطلاعات پولیپ‌های احتمالی و پنهان در ویژگی‌های سطح پایین استفاده می‌شود و اس-ای-ام مجهز به یک لایه پیچشی گرافی غیر محلی جهت استخراج پیکسل‌های محلی و اطلاعات معنایی سراسری از ناحیه پولیپ است [۲۰].

<sup>73</sup>Reverse Attention Module

<sup>74</sup>Global feature map

<sup>75</sup>Parallel partial decoder

<sup>76</sup>Polyp-PVT

<sup>77</sup>CFM

<sup>78</sup>Cascaded Fusion Module

<sup>79</sup>CIM

<sup>80</sup>Camouflage identification module

<sup>81</sup>SAM

<sup>82</sup>Similarity aggregation module

# فصل سوم

## رویکرد پروژه

## ۱-۳ مقدمه

همانطور که قبلتر اشاره شد، مدل پایه که در این پژوهش مورد استفاده قرار گرفته است، مربوط به مقاله [۲] می‌باشد. در این فصل بنا داریم که ابتدا به تشریح و توضیح رویکرد این مدل پردازیم و کارهای صورت گرفته بر روی این مدل را به همراه نتایج آن گزارش کنیم.

این فصل شامل چند زیربخش خواهد بود. زیربخش اول راجع به مدل پایه است. در این زیربخش قصد داریم که مدل استفاده شده در مقاله اصلی و اجزاء مختلف آنرا توضیح دهیم. زیربخش بعدی را به تغییرات انجام شده بر روی مدل پایه اختصاص داده‌ایم. به بیان بهتر، در زیربخش مذکور، قصد داریم که اصلاحات انجام شده و ایده‌هایی که می‌توان برای بهبود مدل انجام داد را تشریح کنیم. این اصلاحات عبارتند از: تغییر در معماری کدگشا، اضافه نمودن تابع زیان جدید، تغییر بهینه‌ساز مورد استفاده در مقاله اصلی.

سپس در ادامه همین فصل، معیارهایی را برای سنجش نتایج معرفی خواهیم کرد، آنها را توضیح خواهیم داد و نتایج اصلاحات انجام شده بر روی مدل را بر اساس همین معیارها در قالب جداول نشان خواهیم داد.

## ۲-۳ مدل پایه

در این پژوهش، ما از ترنس‌یونت استفاده کردیم. ابتدا آزمایش‌ها را با فرایامترهای<sup>۱</sup> ذکر شده در متن مقاله پایه روی مجموعه داده سیناپس<sup>۲</sup> (مجموعه داده مورد استفاده در مقاله پایه) بازتولید کردیم و به نتایج مذکور در مقاله رسیدیم؛ سپس سعی کردیم تا مدل را بر روی مجموعه دادگان ذکر شده در پروژه خود آموزش دهیم. معماری این مدل هم شامل مبدل و هم شبکه پیچشی یو-شکل است و می‌تواند به عنوان یک جایگزین قوی برای قطعه‌بندی تصاویر پزشکی استفاده شود. از یک طرف مبدل، وصله‌های تصویر برآمده از نقشه ویژگی (حاصل از شبکه عصبی پیچشی) را به عنوان دنباله ورودی برای استخراج اطلاعات معنایی سراسری کدگذاری می‌کند. از سوی دیگر کدگشا، ویژگی‌های کدگذاری شده را نمونه‌افزایی می‌کند تا محلی‌سازی دقیق را امکان‌پذیر سازد [۲].

در واقع هم اطلاعات محلی (اطلاعات مکانی) و هم اتصال سراسری (اطلاعات معنایی) به استدلال در

<sup>1</sup>Hyper-parameters

<sup>2</sup>Synapse dataset

مورد روابط بین محتوای تصویر کمک می‌کنند و هر دو برای درک بصری مهم هستند. در طی عمل پیچش یک پنجره لغزان به ورودی اعمال می‌شود و اطلاعات محلی به صورت ذاتی برای محاسبه بازنمایی‌های<sup>۳</sup> جدید جمع می‌شوند. بنابراین، محلی بودن یک ویژگی ذاتی برای شبکه‌های عصبی پیچشی است اما آنها هنوز فاقد اتصال سراسری هستند. در مقابل، مبدل‌ها به خاطر داشتن سازوکار توجه، در مدل‌سازی وابستگی‌های دوربرد در یک دنباله خوب هستند و در واقع توانایی ادراک اطلاعات معنایی را دارند اما در ادراک اطلاعات محلی ضعیف هستند. البته برخی از پژوهش‌ها قبلاً به این هدف کمک کرده‌اند و برای مثال همین معماری ترنس‌یونت از یک نوع معماری ترکیبی بین شبکه‌های عصبی پیچشی و مبدل‌ها استفاده می‌کند تا هم اطلاعات مکانی با وضوح بالا را از ویژگی‌های شبکه‌های عصبی پیچشی و هم اطلاعات معنایی کدگذاری‌شده را از مبدل‌ها دریافت کند [۲۲].

حال که به صورت خلاصه با مدل پایه آشنا شدیم، به توضیح ساختار آن می‌پردازیم. فرض می‌کنیم تصویری  $x \in \mathbb{R}^{H \times W \times C}$  با اندازه  $H \times W$  و تعداد کانالهای  $C$  داده می‌شود. هدف ما پیش‌بینی تصویر ماسک با اندازه  $H \times W$  است. متداول‌ترین راه، آموزش مستقیم یک شبکه عصبی پیچشی (به عنوان مثال شبکه یونت) می‌باشد تا ابتدا ویژگی‌های سطح بالای تصویر با استفاده از کدگذار بازنمایی شوند و سپس با استفاده از کدگشا به وضوح کامل بازگردند. اما برخلاف روش‌های موجود، ما از سازوکار خودتوجهی<sup>۴</sup> موجود در مبدل‌ها در طراحی کدگذار استفاده می‌کنیم. ابتدا در زیربخش اول توضیح کوتاهی راجع به سازوکار توجه می‌دهیم و سپس در زیربخش بعدی نحوه اعمال مستقیم مبدل برای کدگذاری بازنمایی ویژگی‌ها از وصله‌های تصویر تجزیه شده را معرفی می‌کنیم و در نهایت در زیربخش آخر چارچوب کلی ترنس‌یونت شرح داده خواهد شد [۲].

<sup>۳</sup>Representations<sup>۴</sup>Self-attention mechanism

## ۱-۲-۳ سازوکار توجه

سازوکار توجه در واقع روشی است برای تمرکز بر روی بخش‌های مهم تصویر و نادیده گرفتن بخش‌های بی‌اهمیت آن. در این روش، از روی بردارهای ورودی کدگذار سه بردار ساخته می‌شود: بردار پرس‌وجو<sup>۵</sup>، بردار کلید<sup>۶</sup> و بردار مقدار<sup>۷</sup>. این سه بردار از حاصلضرب بردارهای جاسازی با سه ماتریسی که در طی فرآیند آموزش مقدار آنها مشخص شده بدست می‌آیند. در نهایت با استفاده از این سه بردار، ماتریس توجه محاسبه می‌شود. در این ماتریس، هر سطر نشان‌دهنده وزن‌دهی متناظر با یک ورودی است.

خودتوجهی، سازوکار توجهی است که موقعیتهای مختلف یک دنباله را به منظور محاسبه بازنمایی از یک دنباله به یکدیگر مرتبط می‌کند [۵]. این سازوکار مبتنی بر یک حافظه قابل یادگیری با جفت‌های برداری (کلید، مقدار) است. بردار پرس‌وجوی  $Q$  در بردار کلید  $K$  ضرب داخلی می‌شود تا بدین‌صورت امتیاز بدست آید. سپس این ضربهای داخلی با اعمال یک تابع سافت‌مکس<sup>۸</sup> جهت بدست آوردن  $k$  وزن با مجموع یک، نرمال می‌شوند و در نهایت هر بردار مقدار  $V$  در خروجی تابع سافت‌مکس، ضرب می‌شود و بردارهای مقدار وزن‌دار با یکدیگر جمع می‌شوند. برای دنباله ای از  $N$  بردار پرس‌وجو، یک ماتریس خروجی تولید می‌شود و داریم [۱۶]:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (۱-۳)$$

لازم به ذکر است که  $\sqrt{d_k}$  نرمال‌سازی مناسب را فراهم می‌کند و باعث رسیدن به گرادیان‌های پایدارتر می‌شود.

خودتوجهی چندسر<sup>۹</sup> تعمیمی از خودتوجهی است که در آن  $k$  عملیات خودتوجهی را که «هد<sup>۱۰</sup>» نامیده می‌شوند، به صورت موازی اجرا می‌کنیم و خروجی‌های به هم پیوسته آن‌ها را نمایش می‌دهیم [۷].

<sup>۵</sup>Query<sup>۶</sup>Key<sup>۷</sup>Value<sup>۸</sup>Softmax<sup>۹</sup>Multi-head self attention (MSA)<sup>۱۰</sup>Head

## ۲-۲-۳ مبدل به عنوان کدگذار

برای استفاده از مبدل بینایی، که قبلاً در مقدمه فصل دوم توضیح دادیم، باید تصاویر را مسطح کنیم. این کار با شناسه‌سازی<sup>۱۱</sup> (یا جاسازی وصله) یعنی تقسیم هر تصویر به وصله‌هایی دوبعدی با اندازه  $P \times P$  انجام می‌شود.

$$\{x_p^i \in \mathbb{R}^{P \times C} | i = 1, \dots, N\} \quad (۲-۳)$$

که در آن  $N = \frac{H \times W}{P^2}$  تعداد وصله‌های تصویر و  $C$  تعداد کانالهای تصویر است. (طول دنباله ورودی) قبل از وارد کردن این دنباله به لایه‌های توجه چندسر، وصله‌های تصویر مسطح شده با استفاده از یک تابع خطی به فضایی با ابعاد بالاتر (D-بعدی) نگاشت می‌شوند. این کار توسط یک لایه پیچشی منفرد<sup>۱۲</sup> با تابع فعال‌ساز خطی<sup>۱۳</sup> انجام می‌شود. برای کدگذاری اطلاعات مکانی وصله، جاسازی‌های مکانی نیز به جاسازی‌های وصله اضافه می‌شوند تا اطلاعات مکانی (موقعیتی) را به شرح زیر حفظ کنند:

$$z_o = [x_{class}; x_p^1 E; x_p^2 E; \dots; x_p^N E] + E_{pos} \quad (۳-۳)$$

که در آن  $E$ ، ماتریس قابل یادگیری جاسازی وصله و  $E_{pos}$ ، ماتریس جاسازی مکانی می‌باشد. کدگذار مبدل از  $L$  لایه از بلوکهای خودتوجهی چندسر و پرسپترون چند لایه تشکیل شده است. این جاسازی‌ها اکنون به لایه‌های توجه چندسر وارد می‌شوند که در آن سه بردار  $q$ ،  $k$  و  $v$  از هر وصله ورودی استخراج می‌شود و ماتریس توجه بدست می‌آید. در آخر خروجی نهایی یک لایه توجه چندسر به صورت زیر محاسبه می‌شود:

$$z'_l = MSA(LN(z_l - 1)) + z_l - 1 \quad (۴-۳)$$

$$z_l = MLP(LN(z'_l)) + z'_l \quad (۵-۳)$$

در مدل ما از دوازده لایه خودتوجهی چندسر استفاده شده است و هر لایه از دوازده هِد (سر) خودتوجهی تشکیل شده است [۲].

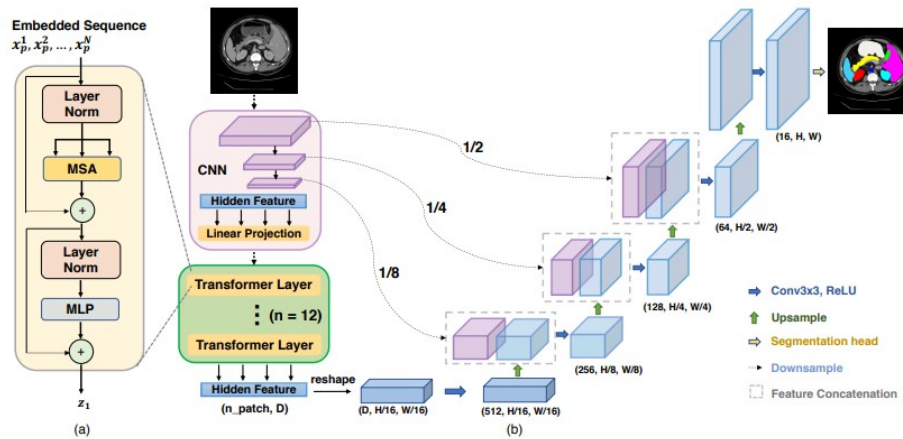
<sup>۱۱</sup>Tokenization

<sup>۱۲</sup>Single convolutin layer

<sup>۱۳</sup>Linear activation function

## ۳-۲-۳ چارچوب کلی ترنس یونت

شکل ۱-۳ معماری ترنس یونت را نشان می دهد. همانطور که مشخص هست، پس از دریافت کدگذاری ها از مبدل  $z_l \in \mathbb{R}^{\frac{HW}{P^2} \times D}$ ، باید آنها را دوباره نمونه افزایی کنیم تا ماسک بدست آید.  $Y \in \mathbb{R}^{H \times W \times K}$  به معنای تعداد کلاس ها می باشد) برای بازیابی ترتیب مکانی، بردار ویژگی کدگذاری شده ابتدا باید از  $\mathbb{R}^{\frac{HW}{P^2} \times D}$  به  $\mathbb{R}^{\frac{H}{P} \times \frac{W}{P} \times D}$  تغییر اندازه دهد.



شکل ۱-۳: معماری ترنس یونت [۲].

اکنون دو روش برای نمونه افزایی وجود دارد:

- نمونه افزایی دوطبی<sup>۱۴</sup>

در اینجا ما به سادگی از یک لایه پیچشی  $1 \times 1$  استفاده می کنیم تا اندازه کانال را از  $D$  به تعداد کلاس های  $K$  کاهش دهیم. سپس، تصویر کدگذاری شده  $z \in \mathbb{R}^{\frac{H}{P} \times \frac{W}{P} \times K}$  به صورت دوطبی تا رسیدن به اندازه کامل ماسک<sup>۱۵</sup>  $Y \in \mathbb{R}^{H \times W \times K}$  نمونه افزایی می شود [۲].

- نمونه افزایی آبشاری<sup>۱۶</sup>

پودمان کاپ<sup>۱۷</sup> بخش مهمی از کدگشا می باشد که ویژگی های کدگذاری شده را نمونه افزایی می کند و آنها را با نقشه های ویژگی شبکه عصبی پیچشی با وضوح بالا ترکیب می کند تا محلی سازی دقیق را امکان پذیر سازد.

<sup>14</sup>Bilinear Upsampling

<sup>15</sup>Full mask size

<sup>16</sup>Cascaded Upsampling (CUP)

<sup>17</sup>CUP



نمونه‌افزایی آبشاری شامل چندین مرحله نمونه‌افزایی با استفاده از لایه‌های پیچشی می‌باشد. پس از تغییر اندازه، تصویر کدگذاری شده با یک نمونه‌افزایی دوخطی با ضریب مقیاس<sup>۱۸</sup> دو و یک لایه پیچشی  $3 \times 3$  به همراه تابع فعال‌سازی رلو<sup>۱۹</sup> برای کاهش اندازه کانال، نمونه‌افزایی می‌شود. در مجموع ما از سه مورد از این بلوکها شامل لایه قطعه‌بندی نهایی (لایه پیچشی  $1 \times 1$  از نمونه برداری دوخطی) برای به دست آوردن ماسک نهایی استفاده می‌کنیم [۲].

در حقیقت پودمان نمونه‌افزای آبشاری دنباله‌ای از عملیات نمونه‌افزایی است که به صورت سلسله مراتبی برای افزایش وضوح تصاویر، اعمال می‌شود و برای تجمیع ویژگی‌ها در سطوح وضوح مختلف از اتصالات پرش استفاده می‌کند که بخش‌های کدگذار و کدگشای شبکه را به همدیگر متصل می‌کند. همانطور که در شکل ۳-۱ مشخص است، اتصالات پرش به هر سه سطح نمونه‌افزایی میانی به جز لایه خروجی، یعنی در مقیاس‌های وضوح  $1/2$ ،  $1/4$  و  $1/8$  اضافه می‌شوند. در نهایت این پودمان، ویژگی‌های سطح پایین از شبکه عصبی پیچشی (اطلاعات مکانی) و ویژگی‌های سطح بالا از مبدل (اطلاعات معنایی) را برای تولید خروجی قطعه‌بندی نهایی جمع می‌کند.

نکته مهم دیگر در معماری ترنس‌یونت، استفاده از مبدل پیچشی ترکیبی می‌باشد. وقتی فقط از مبدل بینایی به عنوان کدگذار استفاده می‌شود به نتایج مناسبی می‌رسیم، اما نمی‌توانیم به وضعیت مطلوب برسیم. بنابراین، در مقاله اصلی [۲]، رزنت‌پنجاه-نسخه‌دو<sup>۲۰</sup> به عنوان یک کدگذار اضافی پیشنهاد شده است. در واقع ما رزنت‌پنجاه و وی-آی-تی را با عنوان آرپنجاه-وی-آی-تی<sup>۲۱</sup> ترکیب می‌کنیم. هر دوی این شبکه‌های ماز<sup>۲۲</sup> روی مجموعه داده ایمیج‌نت-۲۰۱۷<sup>۲۳</sup> از پیش آموزش دیده‌اند [۲، ۲۳].

در حقیقت تصویر به آرپنجاه-وی-آی-تی وارد می‌شود و خروجی جاسازی شده  $\hat{Z}_o \in \mathbb{R}^{P \times P \times D}$ ، مسطح شده  $\hat{Z}_o \in \mathbb{R}^{P^* \times D}$  و وارد لایه جاسازی می‌شود که در بخش قبل توضیح دادیم.

<sup>18</sup>Scale factor

<sup>19</sup>ReLU

<sup>20</sup>ResNet50v2

<sup>21</sup>R50-ViT

<sup>22</sup>Backbone network

<sup>23</sup>ImageNet-21k

در مورد جزئیات پیاده‌سازی باید گفت که اندازه تصویر ورودی و اندازه وصله P به ترتیب  $224 \times 224$

و شانزده تنظیم شده است؛ بنابراین، ما باید چهار بلوک نمونه‌افزایی را به طور متوالی در نمونه افزای آبخاری قرار دهیم تا به وضوح کامل برسیم. تابع زیان مورد استفاده در این معماری مجموع وزن دار کراس آنترپی<sup>۲۴</sup> و دایس<sup>۲۵</sup> می‌باشد و ضرایب را به صورت مساوی  $\lambda = \mu = 0.5$  تنظیم کردیم:

$$loss = \lambda L_{dice} + \mu L_{ce} \quad (3-6)$$

مدل پایه با بهینه‌ساز اس-جی-دی<sup>۲۶</sup> با نرخ یادگیری پیش‌فرض یک صدم، آموزش دیده است. اندازه دسته<sup>۲۷</sup> پیش‌فرض شانزده می‌باشد و تعداد پیش‌فرض تکرارهای<sup>۲۸</sup> آموزشی سی هزار برای همه مجموعه دادگان است. همچنین تعداد دورهای<sup>۲۹</sup> پیش‌فرض برابر با صدوپنجاه می‌باشد.

کد مربوط به مدل پایه به زبان پایتون<sup>۳۰</sup>، با استفاده از چارچوب پایتورچ<sup>۳۱</sup> و در محیط گوگل کولب<sup>۳۲</sup> پیاده‌سازی شده است و همچنین تمام مدلهای این پژوهش که در جهت بهبود مدل پایه بوده‌اند را با یک پردازنده گرافیکی<sup>۳۳</sup> تسلا تی‌چهار با دوازده گیگابایت حافظه آموزش داده‌ایم. علت استفاده از کولب، نیاز پروژه به استفاده از پردازنده گرافیکی با قابلیت کودا<sup>۳۴</sup> بود که به علت نداشتن این مورد از آن استفاده کردیم. قابل ذکر است که استفاده از خدمات پردازنده گرافیکی در این محیط با محدودیت‌هایی همراه بود و همین محدودیت‌ها باعث ایجاد وقفه و طولانی‌شدن آموزش مدل می‌گردید و حتی علی‌رغم بارها تهیه حساب پیشرفته کولب<sup>۳۵</sup> همچنان طولانی‌شدن مدت زمان آموزش محدودیت بزرگی بود.

<sup>24</sup>Cross Entropy loss function

<sup>25</sup>Dice loss function

<sup>26</sup>SGD

<sup>27</sup>Batch size

<sup>28</sup>Iteration

<sup>29</sup>Epoch

<sup>30</sup>Python

<sup>31</sup>Pytorch

<sup>32</sup>Google Colab

<sup>33</sup>Graphics Processing Unit (GPU)

<sup>34</sup>CUDA

<sup>35</sup>Google Colab Pro

## ۳-۳ ایده‌های جدید

در این زیربخش قصد داریم که تمامی اصلاحات صورت گرفته و ایده‌های جدید اعمال شده بر روی مدل پایه را شرح داده و آنها را به تفصیل بیان کنیم. ابتدا در مورد ایده تغییر در تابع زیان بحث خواهیم کرد، سپس ایده جایگزین کردن بهینه‌ساز را مطرح می‌کنیم و در آخر در مورد تغییر معماری کدگشا مطالبی را بیان خواهیم کرد.

## ۳-۳-۱ تغییر در معماری کدگشا

مدل‌های موجود مبتنی بر مبدل، توانایی یادگیری اطلاعات محلی محدودی دارند. کارهای قبلی سعی می‌کردند با قرار دادن لایه‌های پیچش (که در استخراج اطلاعات مکانی قدرتمند هستند) در واحدهای کدگذار یا کدگشای مبدل بر این مشکل غلبه کنند، اما باز هم در ارزیابی‌ها خیلی موفق عمل نمی‌کردند. برای حل این مشکل، ما از یک کدگشای مبتنی بر توجه جدید، به نام گسکید<sup>۳۶</sup> استفاده می‌کنیم.

گسکید از اجزای متعددی تشکیل شده است که مهم‌ترین آنها درگاه‌های توجه<sup>۳۷</sup> و پودمان‌های توجه پیچشی<sup>۳۸</sup> می‌باشند که به ترتیب ویژگی‌های استخراج شده را با اتصالات پرش ترکیب می‌کنند و استخراج اطلاعات مکانی و معنایی را همزمان بهبود می‌بخشند. پژوهش‌ها نشان می‌دهد که مبدل‌هایی که دارای کدگشای گسکید هستند، به طور قابل توجهی عملکرد بهتری نسبت به مدل‌های مبتنی بر شبکه‌های عصبی پیچشی، مبتنی بر مبدل و یا ترکیب این دو دارند.

همچنین نویسندگان مقاله [۱] در مورد نتایج بدست آمده از این معماری در مجموعه دادگان پولیپ ادعا دارند که بهترین روش موجود در حال حاضر را ارائه داده‌اند. در همین راستا ما هم تصمیم گرفتیم برای بهبود مدل به سراغ این معماری برویم و آنرا ارزیابی کنیم. لازم به ذکر است اعلام کنیم نتیجه حاصل، باعث بهبود در تصاویر ماسک پیش‌بینی شد. این نتایج در بخش مربوطه در ادامه همین فصل آورده شده است.

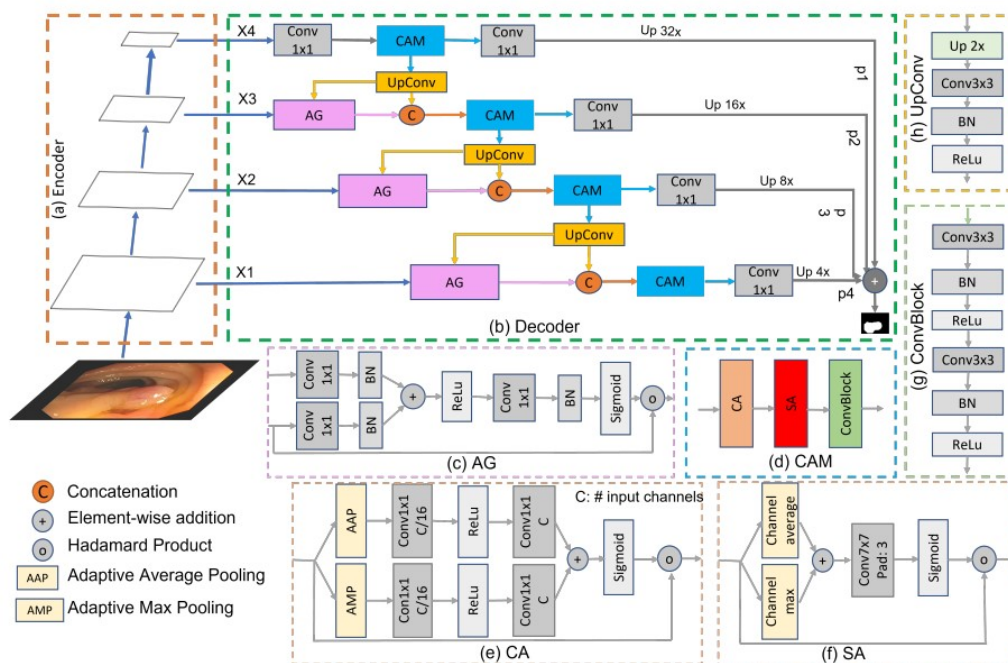
<sup>36</sup>CASCaded Attention DEcoder (CASCADE)

<sup>37</sup>Attention Gate (AG)

<sup>38</sup>Convolutional attention Module (CAM)

از آنجایی که کدگشای پیشنهادی انعطاف‌پذیر است و به راحتی با سایر شبکه‌های مازہ سلسله مراتبی<sup>۳۹</sup> سازگار است، نویسندگان مقاله از مبدل پیچشی ترکیبی<sup>۴۰</sup> که در ترنس‌یونت به کار رفته است، به عنوان کدگذار در این معماری استفاده می‌کنند و معماری حاصل را ترنس‌کسکید<sup>۴۱</sup> می‌نامند [۱].

همانطور که در شکل ۲-۳ نشان داده شده است، معماری کسکید از بلوکهای آپ‌کانو<sup>۴۲</sup> برای نمونه‌افزایی ویژگی‌ها، از درگاه‌های توجه برای ترکیب آبخاری ویژگی‌ها و از بلوکهای کم<sup>۴۳</sup> برای تقویت نقشه‌های ویژگی استفاده می‌کند.



شکل ۲-۳: معماری ترنس‌کسکید [۱].

برای تجمیع ویژگی‌های چند مقیاسی<sup>۴۴</sup>، ابتدا ویژگی‌های نمونه‌افزایی شده از بلوک کدگشای قبلی را با ویژگی‌های اتصالات پرش با استفاده از درگاه توجه ترکیب می‌کند. سپس، ویژگی‌های ترکیب‌شده را با ویژگی‌های نمونه‌افزایی شده از لایه قبلی به هم پیوند<sup>۴۵</sup> می‌دهد. پس از آن، ویژگی‌های به هم پیوسته را با استفاده از بلوک کم، برای گروه‌بندی پیکسل‌هایی با ویژگی مشابه در مناطق مختلف تصویر و کاهش

<sup>۳۹</sup>Hierarchical backbone network

<sup>۴۰</sup>Hybrid CNN-Transformer

<sup>۴۱</sup>TransCASCADE

<sup>۴۲</sup>UpConv

<sup>۴۳</sup>CAM

<sup>۴۴</sup>Multi-scale features

<sup>۴۵</sup>Concatenate

تأثیر ویژگی‌های نامربوط<sup>۴۶</sup> پردازش می‌کند. در نهایت خروجی هر بلوک کم را به یک سر پیش‌بینی<sup>۴۷</sup> ارسال می‌کند و چهار نقشه ویژگی پیش‌بینی شده از مقیاسهای مختلف را برای تولید نقشه قطعه‌بندی<sup>۴۸</sup> نهایی جمع می‌کند و در پایان تابع زیان به این صورت محاسبه می‌شود:

$$loss = \alpha \times loss_{p_1} + \beta \times loss_{p_2} + \gamma \times loss_{p_3} + \zeta \times loss_{p_4} \quad (7-3)$$

که در آن مقادیر  $\alpha$ ,  $\beta$ ,  $\gamma$ ,  $\zeta$  برابر با ۱ می‌باشد و همگی  $loss_{p_1}$  و ... برابر با تابع زیان سرهای پیش‌بینی است.

در ادامه به طور خلاصه بلوک کم و اجزای مهم تشکیل دهنده آن را توضیح خواهیم داد. در گسکید از پودمان‌های توجه پیچشی برای اصلاح نقشه‌های ویژگی استفاده می‌شود. این پودمان‌ها از یک توجه کانالی<sup>۴۹</sup>، یک توجه مکانی<sup>۵۰</sup> و یک بلوک پیچشی تشکیل شده اند.

توجه کانالی مشخص می‌کند که روی کدام ویژگی‌ها باید تمرکز کرد و توجه مکانی تعیین می‌کند که در یک نقشه ویژگی کجا باید تمرکز کرد. بلوک پیچشی که آخرین جزء در کم می‌باشد نیز برای بهبود بیشتر ویژگی‌های ساخته شده توسط توجه کانالی و مکانی استفاده می‌شود [۱].

در شکل ۲-۳ بخش (e)، توجه کانالی نمایش داده شده است. همانطور که مشخص است برای محاسبه این توجه، بر روی بعد مکانی<sup>۵۱</sup> نقشه ویژگی ورودی هر دو عملیات ادغام میانگین و ادغام بیشینه اعمال می‌شود. سپس حاصل هر دو عملیات جداگانه وارد یک لایه پیچشی  $1 \times 1$  با تعداد کانال  $\frac{C}{16}$ ، سپس با عبور از تابع فعالسازی رلو، حاصل وارد لایه پیچشی  $1 \times 1$  با تعداد کانال C می‌شود. بردارهای ویژگی خروجی با استفاده از جمع عناصر ادغام می‌شوند و در آخر تابع سیگموئید<sup>۵۲</sup> روی بردار حاصل اعمال می‌شود.

همانطور که از شکل ۲-۳ بخش (f) مشخص است، توجه مکانی در ادامه توجه کانالی می‌آید پس خروجی توجه کانالی به عنوان ورودی توجه مکانی در نظر گرفته می‌شود و این بار بر روی بعد کانالی<sup>۵۳</sup> نقشه ویژگی ورودی هر دو عملیات ادغام میانگین و ادغام بیشینه اعمال می‌شود. سپس این دو بردار با

<sup>46</sup> Irrelevant features

<sup>47</sup> Prediction head

<sup>48</sup> Segmentation map

<sup>49</sup> Channel attention

<sup>50</sup> Spatial attention

<sup>51</sup> Spatial dimension

<sup>52</sup> Sigmoid function

<sup>53</sup> Channel dimension

هم دیگر ترکیب می‌شوند و حاصل به یک لایه پیمشی  $7 \times 7$  با لایه‌گذاری اضافی<sup>۵۴</sup> سه داده می‌شود و در ادامه تابع سیگموئید هم روی بردار حاصل اعمال می‌شود.

در پایان نویسندگان مقاله [۱] معماری ترکیبی ترنس‌کسکید را با اتخاذ شبکه کدگذار موجود در ترنس‌یونت و کدگشای کسکید معرفی کردند. همانطور که ذکر شد ما نیز از همین معماری برای مدل جدید خود استفاده کردیم و باعث بهبود در عملکرد شد.

### ۲-۳-۳ تغییر تابع زیان

همانطور که قبلاً توضیح داده شد، تابع زیانی که برای آموزش مدل پایه به کار رفته ترکیب وزن دار کراس آنترویی و دایس با ضرایب مساوی  $\frac{1}{2}$  بوده است؛ لذا بر آن شدیم تا به عنوان یک ایده اولیه تابع زیان آی-یو<sup>۵۵</sup> را نیز به این ترکیب اضافه کنیم و نتایج را مشاهده کنیم. یعنی تابع زیان ترکیب وزن‌داری از کراس آنترویی، دایس و آی-یو با ضرایب مساوی  $\frac{1}{3}$  خواهد شد.

بسیاری از پژوهش‌ها در بینایی رایانه با بهینه‌سازی تابعی که به عنوان یک ترکیب خطی وزن دار از توابع زیان چندگانه تعریف می‌شود، آموخته می‌شود و عملکرد نهایی به انتخاب وزنه‌های صحیح (نسبی) برای این توابع زیان حساس است. با اینکه وزن‌های تخصیص داده شده به هر تابع زیان به کاربرد خاص مدل بستگی دارد و تخصیص وزن‌های یکسان همیشه ایده خوبی نیست [۲۴]، اما با این حال کد این تابع زیان را نوشتیم و در آن هم از ترکیب ضرایب متفاوت و هم از ضرایب یکسان استفاده کردیم و برای آموزش مدل پایه و جدید خود از آن استفاده کردیم اما در هر دو صورت هیچ بهبودی در عملکرد نداشت؛ لذا تصمیم گرفتیم که همان ضرایب مساوی را مبنا قرار دهیم.

### ۳-۳-۳ جایگزین کردن بهینه‌ساز

در حالی که بهینه‌سازهای قدیمی مانند آدام دلبیو<sup>۵۶</sup> و اس-جی-دی همچنان مورد استفاده محققان هستند، رویکردهای مختلفی برای کشف خودکار الگوریتم‌های بهینه‌سازی کارآمدتر پیشنهاد شده است. در مقاله جدید [۲۵]، یک تیم تحقیقاتی از گوگل و دانشگاه کالیفرنیا، یک الگوریتم بهینه‌سازی به نام لاین<sup>۵۷</sup> برای آموزش شبکه‌های عصبی ساخته‌اند که با استفاده از یک الگوریتم تکاملی یادگیری ماشین

<sup>۵۴</sup>Padding

<sup>۵۵</sup>IoU loss function

<sup>۵۶</sup>AdamW

<sup>۵۷</sup>EvoLved Sign Momentum (Lion)

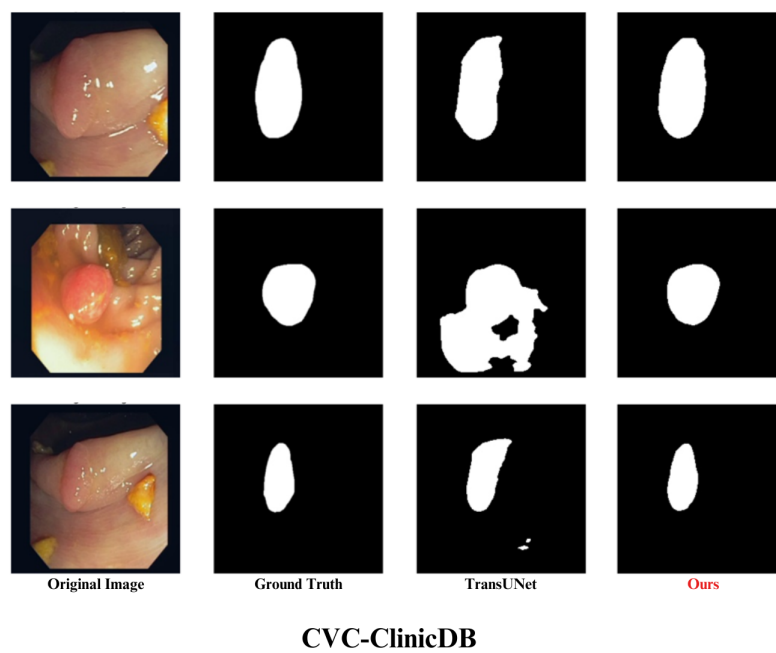
خودکار کشف شده است.

بهینه‌ساز لاین از نظر حافظه کارآمدتر از آدام است به طوریکه به نرخ یادگیری<sup>۵۸</sup> سه تا ده برابر کمتر از آدام نیاز دارد؛ همچنین الگوریتم ساده‌تر با فرآپارامترهای کمتری هم دارد. با استفاده از لاین، محققان چندین مدل را آموزش دادند؛ برای مثال یک مبدل بینایی آموزش دیده توسط این بهینه‌ساز دقت خود را در مجموعه داده ایمیجنت [۲۳]، دو درصد افزایش داده و در عین حال پنج برابر در دوره‌های محاسباتی<sup>۵۹</sup> صرفه جویی کرده بود.

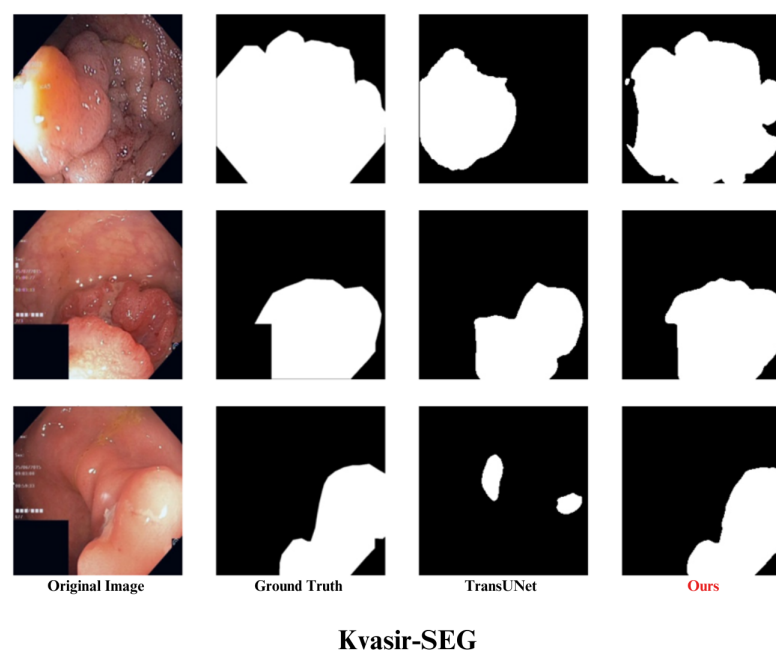
با این توصیفات سعی کردیم تا از بهینه‌ساز اخیرا معرفی شده لاین، در مدل جدید خود استفاده کنیم. البته از آن هم در مدل پایه و هم در مدل جدید استفاده کردیم و با مقایسه به نتایج جالبی رسیدیم به این صورت که استفاده از این بهینه‌ساز در مدل پایه باعث بهبود قابل توجهی شد اما در مدل جدید بهبود مشهودی نداشت. نتایج آزمایش در بخش مربوطه در ادامه همین فصل آورده شده است. و در پایان قصد داریم نتایج را به صورت شهودی نیز نمایش دهیم؛ به همین خاطر نتایج آموزش مدل جدید را به همراه تغییراتی که داده‌ایم روی نمونه‌هایی از مجموعه دادگان پروژه در شکل‌های ۳-۳ و ۴-۳ و ۵-۳ نشان می‌دهیم.

<sup>58</sup> Learning rate

<sup>59</sup> Compute cycles

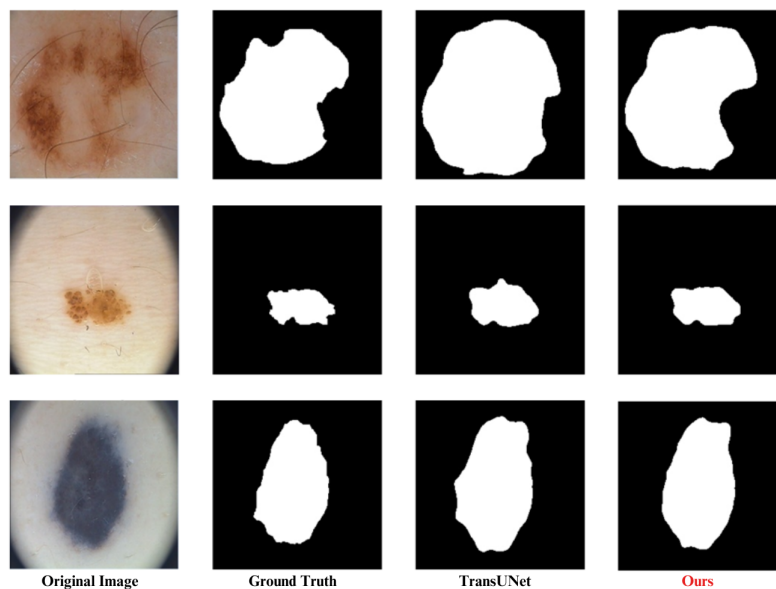


شکل ۳-۳: نتایج کیفی در مجموعه داده CVC-ClinicDB



شکل ۳-۴: نتایج کیفی در مجموعه داده Kvasir-SEG





Ph2

شکل ۳-۵: نتایج کیفی در مجموعه داده Ph2

هر ردیف از این اشکال مربوط به یک نمونه از آن مجموعه داده می‌باشد. ستون اول تصویر، ستون دوم ماسک حقیقی، ستون سوم ماسک پیش‌بینی‌شده توسط مدل پایه (ترنس‌یونت) و ستون چهارم هم مربوط به ماسک پیش‌بینی‌شده توسط مدل بهبود یافته ما (ترنس‌گسکید) می‌باشد. همانطور که به صورت شهودی نیز مشخص است، عملکرد معماری جدید از معماری پایه بهتر بوده است.

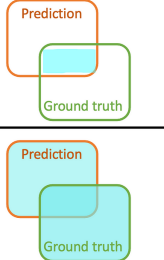
## ۴-۳ معیارها

برای مقایسه مدل پیشنهادی جدید با مدل پایه، از شاخص‌های ارزیابی دایس<sup>۶۰</sup> و آی-اُ-یو<sup>۶۱</sup> استفاده کردیم که به چهار مقدار  $TP$ ،  $TN$ ،  $FP$  و  $FN$  مربوط می‌شوند.

$$Dice = \frac{2 \times TP}{2 \times TP + FP + FN} \quad (۸-۳)$$

$$IoU = \frac{TP}{TP + FP + FN} \quad (۹-۳)$$

به طور دقیق‌تر آی-اُ-یو یا جاکارد<sup>۶۲</sup>، یک معیار ارزیابی پرکاربرد در حوزه قطعه‌بندی تصاویر است که همپوشانی<sup>۶۳</sup> بین ماسک پیش‌بینی‌شده<sup>۶۴</sup> و ماسک حقیقی<sup>۶۵</sup> را اندازه‌گیری می‌کند. نحوه محاسبه این معیار با توجه به شکل ۶-۳ به این صورت است که اشتراک ماسک پیش‌بینی‌شده و ماسک حقیقی بر اجتماع این دو ماسک تقسیم می‌شود.

$$IoU = \frac{\text{Area of overlap}}{\text{Area of union}} = \frac{\text{Prediction} \cap \text{Ground truth}}{\text{Prediction} \cup \text{Ground truth}}$$


شکل ۶-۳: معیار ارزیابی آی-اُ-یو

این معیار برای ارزیابی مدل‌های قطعه‌بندی مهم است، زیرا اندازه‌گیری می‌کند که چگونه مدل می‌تواند اشیاء را از پس‌زمینه‌شان در یک تصویر جدا کند و همچنین می‌تواند عددی بین صفر تا یک را اختیار کند که مقدار صفر به معنی این هست که ماسک پیش‌بینی‌شده و ماسک حقیقی هیچ همپوشانی و اشتراکی ندارند. بیشترین مقدار یعنی یک هم بدین معنا هست که ماسک پیش‌بینی‌شده کاملاً منطبق

<sup>۶۰</sup>Dice

<sup>۶۱</sup>Intersection Over Union (IoU)

<sup>۶۲</sup>Jaccard

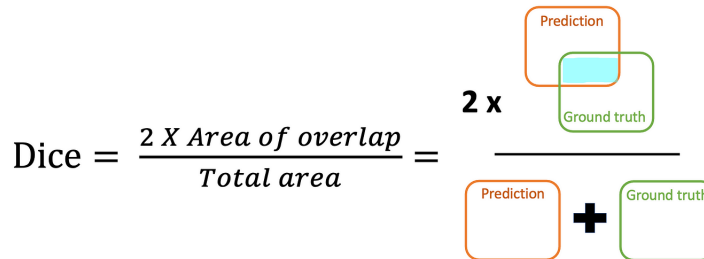
<sup>۶۳</sup>Overlap

<sup>۶۴</sup>Prediction mask

<sup>۶۵</sup>Ground truth mask

بر ماسک حقیقی است.

ضریب دایس<sup>۶۶</sup> هم یکی دیگر از معیارهای ارزیابی رایج برای مدل‌های قطعه‌بندی تصاویر است و مانند معیار آی-اُ-یو شباهت بین ماسک پیش‌بینی‌شده و ماسک حقیقی را اندازه‌گیری می‌کند. این معیار همانطور که در شکل ۳-۷ مشاهده می‌شود به صورت دو برابر اشتراک ماسک پیش‌بینی‌شده و حقیقی تقسیم بر مجموع این دو ماسک محاسبه می‌شود.



$$\text{Dice} = \frac{2 \times \text{Area of overlap}}{\text{Total area}} = \frac{2 \times \text{Area of overlap}}{\text{Prediction} + \text{Ground truth}}$$

شکل ۳-۷: معیار ارزیابی دایس

لازم به ذکر است بیان کنیم، ماتریس سردرگمی<sup>۶۷</sup> جدولی است که عملکرد یک مدل را با مقایسه ماسک‌های پیش‌بینی‌شده آن با ماسک‌های حقیقی خلاصه می‌کند. مقادیر این جدول  $TP^{68}$ ،  $TN^{69}$ ،  $FP^{70}$  و  $FN^{71}$  می‌باشند.  $TP$  و  $TN$  نشان دهنده تعداد پیش‌بینی‌های صحیح هستند، در حالی که  $FP$  و  $FN$  نشان دهنده خطاهای مدل می‌باشند. همانطور که بالاتر دیدیم، این عبارات برای محاسبه معیارهای ارزیابی مانند آی-اُ-یو و دایس استفاده می‌شوند.

<sup>66</sup>Dice coefficient

<sup>67</sup>Confusion matrix

<sup>68</sup>True Positive

<sup>69</sup>True Negative

<sup>70</sup>False Positive

<sup>71</sup>False Negative

## ۵-۳ نتایج

در این زیربخش قصد داریم نتایج مدل‌هایی را که توانستند در تصاویر بهبودی حاصل کنند، بر اساس معیارهای نام‌برده در زیربخش قبلی، در جداول ذکر کنیم. اولین جدول مربوط به نتایج توابع زیان مختلف می‌باشد که با در نظر گرفتن نتایج، می‌توان گفت که ایده اضافه کردن تابع زیان آی-یو باعث بهبود نشده است و در این گام مدل پایه بهتر عمل کرده است؛ لذا از این تابع زیان در ادامه استفاده نخواهد شد.

جدول ۳-۱: نتایج استفاده از توابع زیان مختلف

مدل پایه (TransUNet)		مدل جدید (TransCASCADE)		معیار	مجموعه داده
بدون تابع زیان IoU	با تابع زیان IoU	بدون تابع زیان IoU	با تابع زیان IoU		
۰.۷۸	۰.۸۱	۰.۷۸	۰.۸۱	DSC	CVC-ClinicDB
۰.۶۸	۰.۶۹	۰.۶۹	۰.۶۹	IOU	
۰.۸۸	۰.۹۱	۰.۸۸	۰.۹۰	DSC	Kvasir-SEG
۰.۸۲	۰.۸۴	۰.۸۱	۰.۸۵	IOU	
۰.۹۴	۰.۹۶	۰.۹۴	۰.۹۵	DSC	Ph2
۰.۹۰	۰.۹۲	۰.۹۰	۰.۹۲	IOU	

لازم به ذکر است بیان کنیم، ستون با عنوان «بدون تابع زیان آی-یو» بدین معنا می‌باشد که تابع زیان مدل در این گام همان ترکیب کراس آنترپوی و دایس با ضریب مساوی می‌باشد که قبلاً ذکر شد و در آن از تابع زیان آی-یو استفاده نشده است. همچنین در این جدول مدل پایه همان ترنس‌یونت و مدل جدید ترنس‌کسکید می‌باشد و منظور از معیار دی-اس-سی (DSC) و آی-یو (IOU)، همان معیار دایس و آی-یو می‌باشد که در زیربخش قبلی توضیح دادیم.

در جدول ۳-۲، نتایج مربوط به اعمال بهینه‌سازهای مختلف بر روی مدل پایه و جدید را شاهد هستیم. با توجه به نتایج حاصل از جایگزین کردن بهینه‌ساز لاین به جای اس-جی-دی در مدل پایه و به جای آدام‌دبلیو در مدل جدید، می‌توان گفت که بهینه‌ساز جدید باعث بهبود اندکی در مدل پایه شده است ولی در مورد مدل جدید بهبود مشهودی نداریم؛ با این وجود علی‌رغم بهبود در مدل پایه با استفاده کردن از بهینه‌ساز جدید، نتایج نشان می‌دهد که مدل جدید حتی بدون استفاده کردن از این بهینه‌ساز بهترین نتیجه را داشته است. بنابراین استفاده کردن از بهینه‌ساز جدید در مدل جدید سودی نداشته و در استفاده کردن از آن مختار هستیم ولی با این وجود ما آنرا به عنوان بهینه‌ساز اصلی مدل انتخاب

کردیم.

جدول ۳-۲: نتایج استفاده از بهینه‌سازهای مختلف

مدل جدید (TransCASCADE)		مدل پایه (TransUNet)		معیار	مجموعه داده
با بهینه‌ساز Lion	با بهینه‌ساز SGD	با بهینه‌ساز Lion	با بهینه‌ساز SGD		
۰.۸۱	۰.۸۱	۰.۸۱	۰.۷۸	DSC	CVC-ClinicDB
۰.۶۹	۰.۶۹	۰.۶۸	۰.۶۸	IOU	
۰.۹۱	۰.۹۱	۰.۹۰	۰.۸۸	DSC	Kvasir-SEG
۰.۸۵	۰.۸۴	۰.۸۳	۰.۸۲	IOU	
۰.۹۷	۰.۹۶	۰.۹۶	۰.۹۴	DSC	Ph2
۰.۹۴	۰.۹۲	۰.۹۲	۰.۹۰	IOU	

در آخر در جدول ۳-۳ که آخرین جدول مربوط به مقایسه نتایج می‌باشد، نتیجه حاصل از مدل پایه بدون تابع زیان و بهینه‌ساز جدید را با نتیجه حاصل از مدل جدید با معماری کدگشای متفاوت و بهینه‌ساز جدید مقایسه می‌کنیم. با در نظر گرفتن نتایج، می‌توان نتیجه گرفت که مدل جدید از مدل پایه عملکرد بهتری داشته است.

جدول ۳-۳: نتایج مقایسه مدل پایه و جدید

مدل جدید (TransCASCADE)		مدل پایه (TransUNet)		معیار	مجموعه داده
• بدون تابع زیان IoU • با بهینه‌ساز Lion		• بدون تابع زیان IoU • با بهینه‌ساز SGD			
۰.۸۱		۰.۷۸		DSC	CVC-ClinicDB
۰.۶۹		۰.۶۸		IOU	
۰.۹۱		۰.۸۸		DSC	Kvasir-SEG
۰.۸۵		۰.۸۲		IOU	
۰.۹۷		۰.۹۴		DSC	Ph2
۰.۹۴		۰.۹۰		IOU	

می‌توان گفت در مجموعه داده CVC-ClinicDB، مدل ما به میانگین ۰.۸۱ در معیار دایس و ۰.۶۹ در معیار آی-یو دست می‌یابد که به ترتیب ۳.۸٪ و ۱.۴٪ بیشتر از مدل پایه است.

در مجموعه داده Kvasir-SEG، مدل ما به میانگین ۰.۹۱ در معیار دایس و ۰.۸۵ در معیار آی-یو دست می‌یابد که به ترتیب ۳.۴٪ و ۳.۶٪ بیشتر از مدل پایه است.

در مجموعه داده Ph2، مدل ما به میانگین ۰.۹۷ در معیار دایس و ۰.۹۴ در معیار آی-یو دست می‌یابد که به ترتیب ۳.۲٪ و ۴.۴٪ بیشتر از مدل پایه است.

## فصل چهارم

### پیاده‌سازی سامانه نهایی

در این بخش قصد داریم که در مورد سامانه نهایی پیاده‌سازی شده شرح دهیم. از آنجا که کار کردن با وب بسیار فراگیر شده است و به سادگی می‌تواند در دسترس همگان قرار گیرد، برای سامانه نهایی پروژه، تصمیم گرفتیم که آنرا تحت وب توسعه دهیم و برای این کار از کتابخانه گریدیو<sup>۱</sup> در پایتون استفاده نمودیم.

گریدیو، یک کتابخانه متن‌باز در پایتون است که به شما امکان می‌دهد در سریعترین حالت یک نسخه نمایشی با رابط کاربری ساده و زیبا از مدل یادگیری ماشین خود ایجاد کنید و هر کسی می‌تواند با استفاده از مرورگر خود از آن استفاده کند. در واقع یکی از مزایای آن این است که به شما امکان می‌دهد با برنامه وب که در حال توسعه در جویپتر نوت بوک یا کولب هستید، تعامل داشته باشید. گریدیو همچنین می‌تواند با اکثر چارچوبهای معروف مانند پایتورچ تعامل داشته باشد و از این جهت برای ما مناسب است چرا که ما نیز از چارچوب پایتورچ برای انجام این پروژه استفاده کردیم.

در شکل ۴-۱ نمایی از صفحه وب پیاده‌سازی شده قبل از بارگذاری تصویر نشان داده شده است. همانطور که در تصویر مشهود است، در صفحه اصلی کاربر ابتدا نوع عکس را انتخاب می‌کند. ما در این پژوهش فقط بر روی دو نوع تصویر پولیپ و ضایعه پوستی کار کردیم. بعد از انتخاب نوع تصویر، کاربر می‌تواند تصویر خود را (با هر اندازه‌ای) چه با کشیدن و رها کردن<sup>۲</sup> و چه با انتخاب فایل به صورت محلی بارگذاری کند و سپس دکمه سابمیت<sup>۳</sup> را بفشارد. در این صورت تصویر انتخاب شده بارگزاری خواهد شد و با استفاده از مدل آموزش دیده شده برای آن نوع تصویر، خروجی ماسک پیش بینی شده بعد از چند ثانیه در بخش سمت راست نمایش داده خواهد شد.

در شکل ۴-۲ خروجی سامانه را بعد از بارگذاری تصویر می‌بینیم. حال کاربر پس از مشاهده نتیجه سامانه می‌تواند خروجی را بارگیری کند و همچنین تصمیم بگیرد که آیا خروجی بدست آمده صحیح بوده است یا خیر. در صورتی که تصویر خروجی صحیح بوده است، می‌تواند دکمه «علامت‌گذاری صحیح»<sup>۴</sup> را بفشارد و در غیر این صورت دکمه «علامت‌گذاری غلط»<sup>۵</sup> را بفشارد و در صورتی که مشکلی در انجام عملیات پیش آمده است، دکمه «علامت‌گذاری دیگر»<sup>۶</sup> را بفشارد. در نهایت با کلیک بر روی هر کدام از دکمه‌ها، داده‌های ورودی و خروجی به سروری که در آن نسخه آزمایشی گریدیو اجرا می‌شود، ارسال

<sup>۱</sup>Gradio

<sup>۲</sup>Drag and Drop

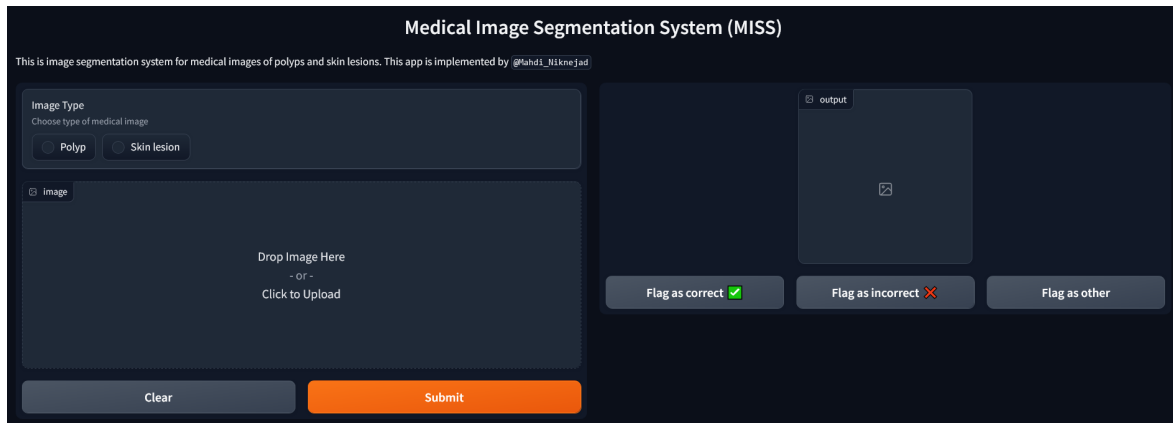
<sup>۳</sup>Submit

<sup>۴</sup>Flag as correct

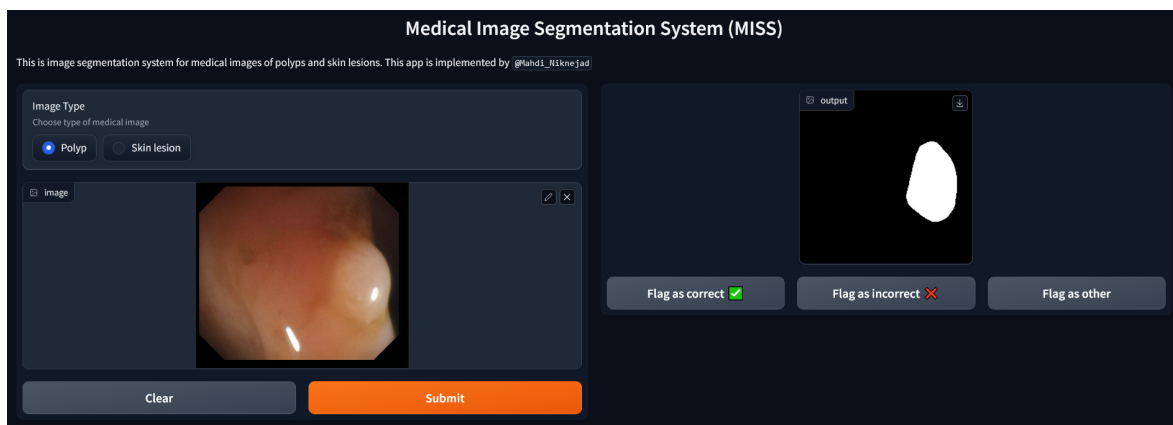
<sup>۵</sup>Flag as incorrect

<sup>۶</sup>Flag as other

می‌شوند و در قالب یک فایل گزارش<sup>۷</sup> سی-اس-وی<sup>۸</sup> ذخیره می‌شوند.



شکل ۴-۱: تصویر سامانه نهایی قبل از بارگذاری تصویر



شکل ۴-۲: تصویر سامانه نهایی بعد از بارگذاری تصویر

<sup>7</sup>Log file

<sup>8</sup>CSV



## فصل پنجم

### جمع‌بندی و نتیجه‌گیری

## ۱-۵ نتیجه‌گیری

در این پژوهش سعی شده بود که با استفاده از یک شبکه ترکیبی پیچشی-مبدلی و تغییر در ساختار کدگشای آن، یک سامانه برای قطعه‌بندی تصاویر پزشکی طراحی شود که به عنوان ورودی تصاویر با دو نوع پولیپ یا ضایعه پوستی دریافت می‌کند و در خروجی تصویر ماسک پیش‌بینی شده را باز می‌گرداند. در فصل دوم با ادبیات معماری شبکه پیچشی یو-شکل (یونت) و مبدل بینایی آشنا شدیم و در ادامه همین فصل به مجموعه دادگان موجود و کارهای مرتبط در این حوزه اشاره شد. در فصل سوم به صورت تفصیلی مقاله ترنس‌یونت مورد بررسی قرار گرفت و جزئیات آن شرح داده شد. سپس در همین فصل به کارهایی که در راستای بهبود عملکرد مقاله اصلی می‌توانست صورت گیرد پرداختیم و نتایج پیاده‌سازی شده را مقایسه کردیم. نهایتاً از مدل بهبودیافته برای ایجاد سامانه استفاده کردیم.

## ۲-۵ کارهای آینده

برای کارهای آینده در این مسأله می‌توانیم از سایر نسخه‌های مبدل بینایی مثل پی-وی-تی، تی-تو-تی<sup>۱</sup> و حتی دیت به عنوان جایگزین برای وی-آی-تی و از نسخه‌های مبدل کارآمدتر نسبت به ترنسفورمر<sup>۲</sup> مثل پریفورمر<sup>۳</sup>، ریفورمر<sup>۴</sup>، کامپکت ترنسفورمر<sup>۵</sup> و یا سوئین ترنسفورمر<sup>۶</sup> استفاده کنیم. همچنین می‌توانیم بر روی داده‌افزایی و پس‌پردازش جهت بهبود کیفیت تصاویر تمرکز کنیم که هر کدام می‌توانند ارزشمند باشند.

---

<sup>1</sup> Tokens-to-Token ViT (T2T-ViT)

<sup>2</sup> Transformer

<sup>3</sup> Preformer

<sup>4</sup> Reformer

<sup>5</sup> Compact Transformer

<sup>6</sup> Swin Transformer

## کتاب نامه

- [1] Rahman, Md Mostafijur and Marculescu, Radu. Medical image segmentation via cascaded attention decoding. in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 6222–6231, 2023.
- [2] Chen, Jieneng, Lu, Yongyi, Yu, Qihang, Luo, Xiangde, Adeli, Ehsan, Wang, Yan, Lu, Le, Yuille, Alan L, and Zhou, Yuyin. Transunet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306, 2021.
- [3] Cao, Hu, Wang, Yueyue, Chen, Joy, Jiang, Dongsheng, Zhang, Xiaopeng, Tian, Qi, and Wang, Manning. Swin-unet: Unet-like pure transformer for medical image segmentation. in European Conference on Computer Vision, pp. 205–218. Springer, 2022.
- [4] Pan, Shaoming, Liu, Xin, Xie, Ningdi, and Chong, Yanwen. Eg-transunet: Enhanced and guided u-net with transformer for biomedical image segmentation. 2022.
- [5] Vaswani, Ashish, Shazeer, Noam, Parmar, Niki, Uszkoreit, Jakob, Jones, Llion, Gomez, Aidan N, Kaiser, Łukasz, and Polosukhin, Illia. Attention is all you need. Advances in neural information processing systems, 30, 2017.
- [6] Ronneberger, Olaf, Fischer, Philipp, and Brox, Thomas. U-net: Convolutional networks for biomedical image segmentation. in Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18, pp. 234–241. Springer, 2015.

- [7] Dosovitskiy, Alexey, Beyer, Lucas, Kolesnikov, Alexander, Weissenborn, Dirk, Zhai, Xiaohua, Unterthiner, Thomas, Dehghani, Mostafa, Minderer, Matthias, Heigold, Georg, Gelly, Sylvain, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020.
- [8] Devlin, Jacob, Chang, Ming-Wei, Lee, Kenton, and Toutanova, Kristina. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.
- [9] Bernal, Jorge, Sánchez, F Javier, Fernández-Esparrach, Gloria, Gil, Debora, Rodríguez, Cristina, and Vilariño, Fernando. Wm-dova maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians. *Computerized medical imaging and graphics*, 43:99–111, 2015.
- [10] Jha, Debesh, Smedsrud, Pia H, Riegler, Michael A, Halvorsen, Pål, de Lange, Thomas, Johansen, Dag, and Johansen, Håvard D. Kvasir-seg: A segmented polyp dataset. in *MultiMedia Modeling: 26th International Conference, MMM 2020, Daejeon, South Korea, January 5–8, 2020, Proceedings, Part II* 26, pp. 451–462. Springer, 2020.
- [11] Mendonça, Teresa, Ferreira, Pedro M, Marques, Jorge S, Marcal, André RS, and Rozeira, Jorge. Ph 2-a dermoscopic image database for research and benchmarking. in *2013 35th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*, pp. 5437–5440. IEEE, 2013.
- [12] Zhou, Zongwei, Rahman Siddiquee, Md Mahfuzur, Tajbakhsh, Nima, and Liang, Jianming. Unet++: A nested u-net architecture for medical image segmentation. in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings* 4, pp. 3–11. Springer, 2018.

- [13] Oktay, Ozan, Schlemper, Jo, Folgoc, Loic Le, Lee, Matthew, Heinrich, Mattias, Misawa, Kazunari, Mori, Kensaku, McDonagh, Steven, Hammerla, Nils Y, Kainz, Bernhard, et al. Attention u-net: Learning where to look for the pancreas. arXiv preprint arXiv:1804.03999, 2018.
- [14] Qin, Xuebin, Zhang, Zichen, Huang, Chenyang, Dehghan, Masood, Zaiane, Osmar R, and Jagersand, Martin. U2-net: Going deeper with nested u-structure for salient object detection. *Pattern recognition*, 106:107404, 2020.
- [15] Lin, Ailiang, Chen, Bingzhi, Xu, Jiayu, Zhang, Zheng, Lu, Guangming, and Zhang, David. Ds-transunet: Dual swin transformer u-net for medical image segmentation. *IEEE Transactions on Instrumentation and Measurement*, 71:1–15, 2022.
- [16] Touvron, Hugo, Cord, Matthieu, Douze, Matthijs, Massa, Francisco, Sablayrolles, Alexandre, and Jégou, Hervé. Training data-efficient image transformers & distillation through attention. in *International conference on machine learning*, pp. 10347–10357. PMLR, 2021.
- [17] Liu, Ze, Lin, Yutong, Cao, Yue, Hu, Han, Wei, Yixuan, Zhang, Zheng, Lin, Stephen, and Guo, Baining. Swin transformer: Hierarchical vision transformer using shifted windows. in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4784–4793, 2021.
- [18] Wang, Wenhai, Xie, Enze, Li, Xiang, Fan, Deng-Ping, Song, Kaitao, Liang, Ding, Lu, Tong, Luo, Ping, and Shao, Ling. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 568–578, 2021.
- [19] Wang, Wenhai, Xie, Enze, Li, Xiang, Fan, Deng-Ping, Song, Kaitao, Liang, Ding, Lu, Tong, Luo, Ping, and Shao, Ling. Pvt v2: Improved baselines with pyramid vision transformer. arXiv preprint arXiv:2106.13797, 2021.

- [20] Bo, Dong, Wenhai, Wang, Deng-Ping, Fan, Jinpeng, Li, Huazhu, Fu, and Ling, Shao. Polyp-pvt: Polyp segmentation with pyramidvision transformers, 2023.
- [21] Fan, Deng-Ping, Ji, Ge-Peng, Zhou, Tao, Chen, Geng, Fu, Huazhu, Shen, Jianbing, and Shao, Ling. Prnet: Parallel reverse attention network for polyp segmentation. in International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 102–111. Springer, 2020.
- [22] Hu, Han, Chu, Xiangxiang, Xu, Chang, Zhang, Bo, and Wei, Yichen. Localvit: Bringing locality to vision transformers, 2021.
- [23] Russakovsky, Olga, Deng, Jia, Su, Hao, Krause, Jonathan, Satheesh, Sanjeev, Ma, Sean, Huang, Zhiheng, Karpathy, Andrej, Khosla, Aditya, Bernstein, Michael, et al. Imagenet large scale visual recognition challenge. International journal of computer vision, 115:211–252, 2015.
- [24] Groenendijk, Rick, Karaoglu, Sezer, Gevers, Theo, and Mensink, Thomas. Multi-loss weighting with coefficient of variations. in Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp. 1469–1478, 2021.
- [25] Chen, Xiangning, Liang, Chen, Huang, Da, Real, Esteban, Wang, Kaiyuan, Liu, Yao, Pham, Hieu, Dong, Xuanyi, Luong, Thang, Hsieh, Cho-Jui, et al. Symbolic discovery of optimization algorithms. arXiv preprint arXiv:2302.06675, 2023.

# واژه‌نامه‌ی فارسی به انگلیسی

بهترین روش موجود در حال حاضر . state-of-the-art	آ
بازشناسی تصویر . . . . . Image recognition	آندوسکوپی . . . . . Endoscopy
بازنمایی . . . . . Representation	اندازه دسته . . . . . Batch size
پ	الحاق . . . . . Concatenation
پایتون . . . . . Python	اتصالات باقیمانده . . . . . Residual connections
پایتورچ . . . . . Pytorch	اتصالات پرش . . . . . Skip connections
پردازنده گرافیکی Graphics Processing Unit	ادغام بیشینه . . . . . Max pooling
پرس‌وجو . . . . . Query	اطلاعات معنایی سراسری . . . . . Global context
پودمان توجه معکوس . . . . . Reverse attention	اندام . . . . . Organ
module	از پیش آموزش دیده . . . . . Pre-trained
پیچیدگی محاسباتی نمایی درجه دو Quadratic complexity	ب
پردازش زبان طبیعی . . . . . Natural Language	بینایی رایانه . . . . . Computer Vision
Processing	بهینه‌ساز . . . . . Optimizer
پرسپترون چندلایه Multi Layer Perceptron	بلوک باقیمانده U-شکل . Residual U-Block

Trainable linear transformation	پیش‌زمینه . . . . .
projection	پس‌زمینه . . . . .
Object detection . . . . .	پودمان توجه پیچشی . . . . .
Sigmoid function . . . . .	Attention module
Iteration . . . . .	پولیپ . . . . .
ج	پوست‌شناسی . . . . .
Patch embedding . . . . .	ت
Position embedding . . . . .	تشخیص به کمک رایانه . . . . .
Token embedding . . . . .	diagnosis
چ	توجه سراسری . . . . .
Framework . . . . .	توجه مبتنی بر پنجره . . . . .
خ	attention
Self-attention . . . . .	توجه پنجره لغزان
Multi-head self . . . . .	Spatial reduction . . . . .
attention	attention
د	توجه کانالی . . . . .
Classification . . . . .	Spatial attention . . . . .
Epoch . . . . .	توجه مکانی . . . . .
Split . . . . .	تابع زیان . . . . .
دوبخشی	ترجمه ماشینی
Attention gate . . . . .	تغییر اندازه و شکل . . . . .
درگاه توجه	Reshape . . . . .
دنباله به دنباله . . . . .	



Medical Image . . . . . قطعه‌بندی تصاویر پزشکی	Dermoscopy . . . . . درموسکوپی
Segmentation	س
Semantic segmentation . . . . . قطعه‌بندی معنایی	Attention mechanism . . . . . سازوکار توجه
ک	Prediction head . . . . . سر پیش‌بینی
Parallel partial . . . . . کدگشای جزئی موازی	ش
decoder	Convolutional Neural شبکه عصبی پیچشی
Encoder . . . . . کدگذار	Network
Decoder . . . . . کدگشا	Feedforward . . . . . شبکه پیچشی پیش‌خور
Colonoscopy . . . . . کولونوسکوپی	convolutional network
گ	Hierarchical . . . . . شبکه مازہ سلسله مراتبی
Google colab . . . . . گوگل کولب	backbone network
Gradio . . . . . گریدیو	Token . . . . . شناسه
Stride . . . . . گام	Tokenization . . . . . شناسه سازی
ل	Semantic gap . . . . . شکاف معنایی
Padding . . . . . لایه‌گذاری	ض
م	Scale factor . . . . . ضریب مقیاس
Metric . . . . . معیار	Reduction factor . . . . . ضریب کاهش
Confusion Matrix . . . . . ماتریس سردرگمی	Lesion . . . . . ضایعه
Mask . . . . . ماسک	ف
Ground truth mask . . . . . ماسک حقیقی	Hyper-parameter . . . . . فراپارامتر
	ق

Learning rate . . . . . نرخ یادگیری	Prediction mask . . . . . ماسک پیش‌بینی شده
9	Receptive field . . . . . میدان تأثیر
Patch . . . . . وصله	Flattened . . . . . مسطح
Non-overlapped . . . . . وصله‌های غیرهم‌پوشان	Localization . . . . . محلی‌سازی
patches	Contracting path . . . . . مسیر کاهشی
Long-range . . . . . وابستگی‌های دوربرد	Expanding path . . . . . مسیر گسترشی
dependencies	Transformer . . . . . مبدل
Resolution . . . . . وضوح	Vision Transformer . . . . . مبدل بینایی
Multi-scale feature . . . . . ویژگی چندمقیاسی	Hybrid . . . . . مبدل پیچشی ترکیبی
Irrelevant feature . . . . . ویژگی نامربوط	CNN-Transformer
5	Dataset . . . . . مجموعه داده‌گان
Correlation . . . . . همبستگی	Intrinsic . . . . . محلی بودن ذاتی عملیات پیچشی
Overlap . . . . . هم‌پوشانی	locality of convolution operation
ی	ن
Deep learning . . . . . یادگیری عمیق	Upsampling . . . . . نمونه‌افزایی
	Downsampling . . . . . نمونه‌کاهی
	Segmentation map . . . . . نقشه قطعه‌بندی
	Feature map . . . . . نقشه ویژگی
	Layer normalization . . . . . نرمال‌سازی لایه

# واژه‌نامه‌ی انگلیسی به فارسی

A	پودمان توجه پیچشی . . . . . Convolutional
	Attention Module
Attention Mechanism . . . . . سازوکار توجه	
Attention gate . . . . . درگاه توجه	مسیر کاهشی . . . . . Contracting path
B	الحاق . . . . . Concatenation
Batch size . . . . . اندازه دسته	توجه کانالی . . . . . Channel attention
Background . . . . . پس‌زمینه	دسته‌بندی . . . . . Classification
C	D
Computer-aided . . . . . تشخیص به کمک رایانه	کدگشا . . . . . Decoder
diagnosis	مجموعه دادگان . . . . . Dataset
Convolutional Neural شبکه عصبی پیچشی	ایادگیری عمیق . . . . . Deep learning
Network	درموسکوپي . . . . . Dermoscopy
Computer Vision . . . . . بینایی رایانه	پوست‌شناسی . . . . . Dermatology
Colonoscopy . . . . . کولونوسکوپي	نمونه‌کاهی . . . . . Downsampling
Confusion matrix . . . . . ماتریس سردرگمی	E
Correlation . . . . . همبستگی	کدگذار . . . . . Encoder

Endoscopy . . . . . آندوسکوپی	Hyper-parameter . . . . . فراپارامتر
Epoch . . . . . دور	Hierarchical . . . . . شبکه مازہ سلسله مراتبی
Expanding path . . . . . مسیر گسترشی	backbone network
F	I
Framework . . . . . چارچوب	Intrinsic . . . . . محلی بودن ذاتی عملیات پیچشی
Feature map . . . . . نقشه ویژگی	locality of convolution operation
Foreground . . . . . پیش‌زمینه	Iteration . . . . . تکرار
Flattened . . . . . مسطح	Irrelevant feature . . . . . ویژگی نامربوط
Feedforward . . . . . شبکه پیچشی پیش‌خور	Image recognition . . . . . بازشناسی تصویر
convolutional network	L
G	Long-range . . . . . وابستگی‌های دوربرد
Global context . . . . . اطلاعات معنایی سراسری	dependencies
Graphics Processing Unit پردازنده گرافیکی	Lesion . . . . . ضایعه
Google colab . . . . . گوگل کولب	Loss function . . . . . تابع زیان
Gradio . . . . . گریدیو	Localization . . . . . محلی‌سازی
Ground truth mask . . . . . ماسک حقیقی	Layer normalization . . . . . نرمال‌سازی لایه
Global attention . . . . . توجه سراسری	Learning rate . . . . . نرخ یادگیری
H	M
Hybrid . . . . . مبدل پیچشی ترکیبی	Medical Image . . . . . قطعه‌بندی تصاویر پزشکی
CNN-Transformer	Segmentation
	Metric . . . . . معیار

Mask . . . . . ماسک	Project metric . . . . . معیار پروژه
Max pooling . . . . . ادغام بیشینه	Prediction mask . . . . . ماسک پیش‌بینی شده
Multi-head . . . . . خودتوجهی چندسر	Prediction head . . . . . سر پیش‌بینی
self-attention	Parallel partial . . . . . کدگشای جزئی موازی
Multi layer perceptron . . . . . پرسپترون چندلایه	decoder
Multi scale feature . . . . . ویژگی چند مقیاسی	Patch . . . . . وصله
N	Patch embedding . . . . . جاسازی وصله
Natural Language . . . . . پردازش زبان طبیعی	Position embedding . . . . . جاسازی موقعیت
Processing	Padding . . . . . لایه‌گذاری
Non-overlapped . . . . . وصله‌های غیرهمپوشان	Polyp . . . . . پولیپ
patches	Pre-trained . . . . . از پیش آموزش دیده
Neural Machine Translation . . . . . ترجمه ماشینی	Q
O	Query . . . . . پرس‌وجو
Organ . . . . . اندام	Quadratic . . . . . پیچیدگی محاسباتی نمایی درجه دو
Optimizer . . . . . بهینه‌ساز	complexity
Object detection . . . . . تشخیص اشیاء	R
Overlap . . . . . هم‌پوشانی	Resolution . . . . . وضوح
P	Representation . . . . . بازنمایی
Python . . . . . پایتون	Reverse attention . . . . . پودمان توجه معکوس
Pytorch . . . . . پایتورچ	module
	Reduction factor . . . . . ضریب کاهش

Receptive field . . . . . میدان تأثیر	Scale factor . . . . . ضریب مقیاس
Residual connection . . . . . اتصال باقیمانده	Skip connection . . . . . اتصال پرش
Residual U-Block . . . . . بلوک باقیمانده U-شکل	Split . . . . . دوبخشی
Reshape . . . . . تغییر اندازه و شکل	T
S	Transformer . . . . . مبدل
Segmentation map . . . . . نقشه قطعه‌بندی	Token . . . . . شناسه
Sequence-to-sequence . . . . . دنباله به دنباله	Tokenization . . . . . شناسه‌سازی
Self-attention . . . . . سازوکار توجه	Token embedding . . . . . جاسازی شناسه
Sliding window attention . . . . . توجه پنجره لغزان	Trainable linear . . . . . تبدیل خطی قابل یادگیری
Spatial reduction . . . . . توجه کاهشی مکانی	projection
attention	U
Stride . . . . . گام	Upsampling . . . . . نمونه‌افزایی
Semantic gap . . . . . شکاف معنایی	V
State-of-the-art . . . . . بهترین روش موجود در حال حاضر	Vision Transformer . . . . . مبدل بینایی
Semantic segmentation . . . . . قطعه‌بندی معنایی	W
Spatial attention . . . . . توجه مکانی	Window-based . . . . . توجه مبتنی بر پنجره
Sigmoid function . . . . . تابع سیگموئید	attention

# Abstract

Rapid advances in the field of medical imaging are revolutionizing medicine. For example, the diseases diagnosis with the help of computers, where the segmentation of medical images plays an important role, has become more accurate. Although CNN-based methods have achieved excellent performance in recent years, but due to the intrinsic locality of convolution operations, they cannot learn explicit global and long-range semantic information well. Given the increased interest in self-attention mechanisms in computer vision and their ability to overcome this problem, the TransUNet architecture was proposed as the first medical image segmentation framework using Vision Transformer as a strong encoder in a U-shaped architecture.

TransUNet achieves good results compared to different architectures; therefore, in this project, we use it as the base model that has a hybrid CNN-Transformer architecture. this architecture is able to leverage both detailed high-resolution spatial information from CNN features and the global context encoded by Transformers. All experiments are conducted on Kvasir-SEG, CVC-ClinicDB and Ph2 datasets. First, we reproduce the results in the original paper, and then we proceed to improve the architecture by making appropriate changes and check the results. Some of these changes have been successful and others have been unsuccessful. Finally, we created a web-based system based on the new architecture. Code is available at : <https://github.com/mnn59/BSc>

## Key Words:

Deep Learning, Computer Vision, Medical Image Segmentation, UNet, Attention mechanism, Transformer, Colonoscopy, Dermoscopy