Genome analysis

Intra-bin structural variant segmentation for whole-genome sequencing data using U-net

Yao-zhong Zhang [1,*], Seiya Imoto [1], Satoru Miyano [1] and Rui Yamaguchi [1,*]

[1] Institute of Medical Science, the University of Tokyo, Tokyo, 108-0071, Japan

[*] To whom correspondence should be addressed.

Abstract

Motivation: Structure variants (SVs) are complex genomic alterations with affected genome sizes larger than 50 nucleotides. They have been found in both healthy individuals and patients, which contribute to phenotypic traits and are highly correlated with diseases, such as cancer. With the advance of sequencing technologies, more precise profiling of SVs becomes available. It is still a challenging task to determine SVs in nucleotide-resolution with satisfying both high accuracy and coverage for a single sample.

Result: In this paper, we proposed a novel nucleotide-resolution SV detection method for determining SV fragment inside of a bin with base-pair read-depth (RD) information. Distinguished from traditional RD based SV detectors, which usually require grouping and smoothing RD signals at the scale of a bin, we use a U-net model to directly process base-pair RD signals inside of a bin and learn feature representations for intra-bin SV segmentation. The proposed method can be used solely for searching target SVs with RD signature or integrated with previous bin-based SV detection methods to exceed the bin-size limitation. We did a systematic evaluation of the method using WGS data from 1000 Genomes Project. In both single-sample and cross-sample experiments, the U-net model achieves significantly better performances on detecting intra-bin SV fragments, when compared with convolutional neural networks (CNN). Furthermore, the model shows a good ability to detect specific SV subtypes, such as ALU deletion.

Availability: The source code is available at https://github.com/yaozhong/deepIntraSV

Contact: yaozhong@ims.u-tokyo.ac.jp

Supplementary information: Supplementary data are available at https://github.com/yaozhong/deepIntraSV

1 Introduction

Structural variants (SVs), including copy number variations (CNVs), are genomic alterations larger than 50 base pairs (bp). When compared with single nucleotide polymorphisms (SNPs), the larger size of SVs makes them more likely to alter genomic structures and have functional consequences. In many diseases, such as Alzheimer's disease and many types of cancer, SVs have been found as important contributors [1, 2, 3, 4]. To identify SV patterns associated with diseases, complete discovery and accurate genotyping of SVs is the first and foremost step.

Currently, there are mainly two types of platforms used for SV detection and genotyping. One is array based, such as array comparative genomic hybridization (aCGH) and SNP array, which is relatively cheap. The other one is next-generation sequencing (NGS) based, which provides higher resolution of SV profiles. For NGS data, many bioinformatics methods have been developed, which utilize different characters of reads related to SVs. Those methods can be categorized into four types according to used read information: *read depthāŁŁ(RD)*(or read count) [5], *paired-end mapping (PEM)* [6], *split-read (SR)* [7] and *assembly-based (AS)* [8]. A RD based approach assumes the number of reads in a genomic region is proportional to the number of times the region appears in the sample. In general, it fits a random distribution (e.g., negative binomial) to read counts and finds regions with abnormal read counts according to the fitted model. It detects deletion and duplication and is commonly used to detect CNV. PEM uses span and orientation of paired-end reads, which can detect several forms of SVs, such as inversion and tandem duplication, beside deletion and duplication. SR can find nucleotide-resolution breakpoints of SVs based on the observation that an adjoining short-read sequence is separately aligned in different coordinates of a reference genome. AS is a general approach that first does de novo assembly of sample genome based on sequenced reads and then compares the assembled genome with reference genome for structural differences. Theoretically, all types of SVs can be detected by an AS method. In practice, it usually requires high-depth of coverage for genome assembly and has more computational cost. Those SV detection methods vary in both sensitivity and specificity. Methods that integrate more than one type of read signals are also developed. For example, DELLY [6] integrated both PE and SR for SV detection. In this work, we focus on the RD based approach and aim to exceed the resolution of bin signals.

On the other hand, in recent years, deep learning has achieved impressive results for several tasks in computational biology, such as

predicting sequence specificity of binding [9, 10]. For deep learning applications related to SV detection, Poplin et al. proposed DeepVariant [11], which uses a well-calibrated convolutional neural network (CNN) to call SNP and small-indel variants based on image pileups around putative variant sites. Eghbal-zadeh et al. [12] used SNP-array genomic data and applied deep neural network with attention mechanism for breakpoint detection. Those works formulate the detection task as a classification problem and trained deep networks to predict SV labels.

In this paper, we focus a task of finding somatic SV fragments inside of a bin using base-pair RD signals, which refers to as "intra-bin SV segmentation". Usually, base-pair RD signals are too noisy to be directly used. In previous RD-based approaches, non-overlapping bins with fixed-length are applied to group base-pair RD signals and calculate bin-based RD signals for SV detection. Therefore, the resolution of breakpoints detected by those approaches is naturally limited by the bin-size. For an SV inside of a bin, it is easy to be overwhelmed. To conquer this limitation, we propose to use a U-net model for the task. The U-net [13] is the state-of-the-art deep learning model widely used in many image semantic segmentation tasks. The encode-and-decode architecture makes it possible to directly process noisy base-pair RD signals of whole-genome sequencing alignment data. Meanwhile, instead of formulating as a classification task, we treat the problem as a segmentation task, which can provide higher-resolution of breakpoints inside of a bin. The proposed method can be used solely for searching target SVs inside of a bin or integrated with previous bin-based SV detection methods to exceed the bin-size limitation.

We evaluated model performance on real WGS data and performed systematic experiments for intra-sample and cross-sample applications. The experiment results demonstrate that a U-net model can be trained to learn base-pair RD signatures of annotated SV segments inside of a bin. When compared with a CNN model, the U-net achieves significantly better performance and produces more integrated segments. In addition, the U-net model shows a good ability to detect SV subtypes related to mobile elements, such as ALU deletion. We also evaluate the segmentation results in a simplified SV classification task. The U-net model also achieves better AUC score, when compared with SVM.

2 Methods

In this section, we formulate the intra-bin SV segmentation task and describe it in an integrated pipeline. Then, we describe the U-net model in detail for the segmentation task.

2.1 Overall pipeline

We first describe a pipeline, in which the proposed model is supposed to be used. The overall pipeline is described in Figure 1. It is a two-stage pipeline, which consists of coarse-and-global search part and fine-and-local segmentation part. For a sample, the first stage applies bin-based SV detection using previous RD-based method. Base-pair RD signals are grouped by a fixed-length bin (e.g., 1000 bp) as the basic unit for SV detection. In this stage, the whole-genome is globally screened and a list of bin-resolution breakpoint candidates are generated. In the second stage, regions surrounding the breakpoint candidates are zoomed-in with nucleotide-resolution. An intra-bin segmentation model is applied to find SV fragments inside of the regions and acquires higher-resolution breakpoints. In the following part, we focus on the second stage of the intra-bin segmentation.

2.2 Problem definition

Formally, give read-depth vector $X = \{d_1, d_2, ..., d_l\}$ for a target l-bp region, we aim to find its corresponding segmentation $Y = \{m_1, m_2, ..., m_l\}$, where $m_i \in \{0, 1\}$ indicates whether local coordinate i is overlapped with any SV. It is a challenging task, due to the fact that boundary signal is noisy and ambiguous.

2.3 U-net model

U-net [13] is a deep neural network featuring with a U-shape architecture. It has achieved state-of-the-art performance in many image semantic segmentation tasks. We propose to apply U-net model to genome field for SV detection. The structure of the U-net is described in Figure 2(c). It basically is an encode-and-decode architecture, in which left-U part encodes and right-U part decodes. The left-U contracts input signals through repeatedly applying convolutions, each followed by a rectified linear unit (ReLU), batch normalization (BN) and max-pooling operation. The right-U expands the down-sampled feature map by up-convolution and concatenate the correspondingly feature map from the left-U part. Each concatenated feature map is followed by two additional convolutions with ReLU and BN. At the output layer, 1x1 convolution is applied with sigmoid activation to predict mask vectors. Different from a classic CNN model, there is no fully connection layer in a U-net and tensors before down-sampling are concatenated with the up-sampling tensors to keep the position information. In applying U-net to the intra-bin SV segmentation task, we have to guarantee the output length exactly the same as the input length for locating breakpoints in the reference genome. In our U-net model, we use padded convolutions and select proper max-pooling pool size to avoid inconsistency of hidden numbers in the correspondence level of the network. There is no cropped feature map in the right-U part. The output layer is activated by a softmax function. We transform the value above the threshold of 0.5 to be 1, and others to be 0.

2.3.1 Training

To train a U-net model, we collect WGS data with SV annotation. For each SV, we generate regions (bins) with a fixed-length that contain breakpoint(s) of left-end or right-end. The breakpoint is not required in the centric position of a bin. We take a random shift to each breakpoint as the start position of the bin. For each breakpoint containing bin, SV overlapping coordinates are masked with 1, while the rest are masked with 0. We generate the same number of background regions without overlapping any SV as negative samples.

We use Dice similarity coefficient (DSC) as the loss function, and Adam [14] as the optimizer.

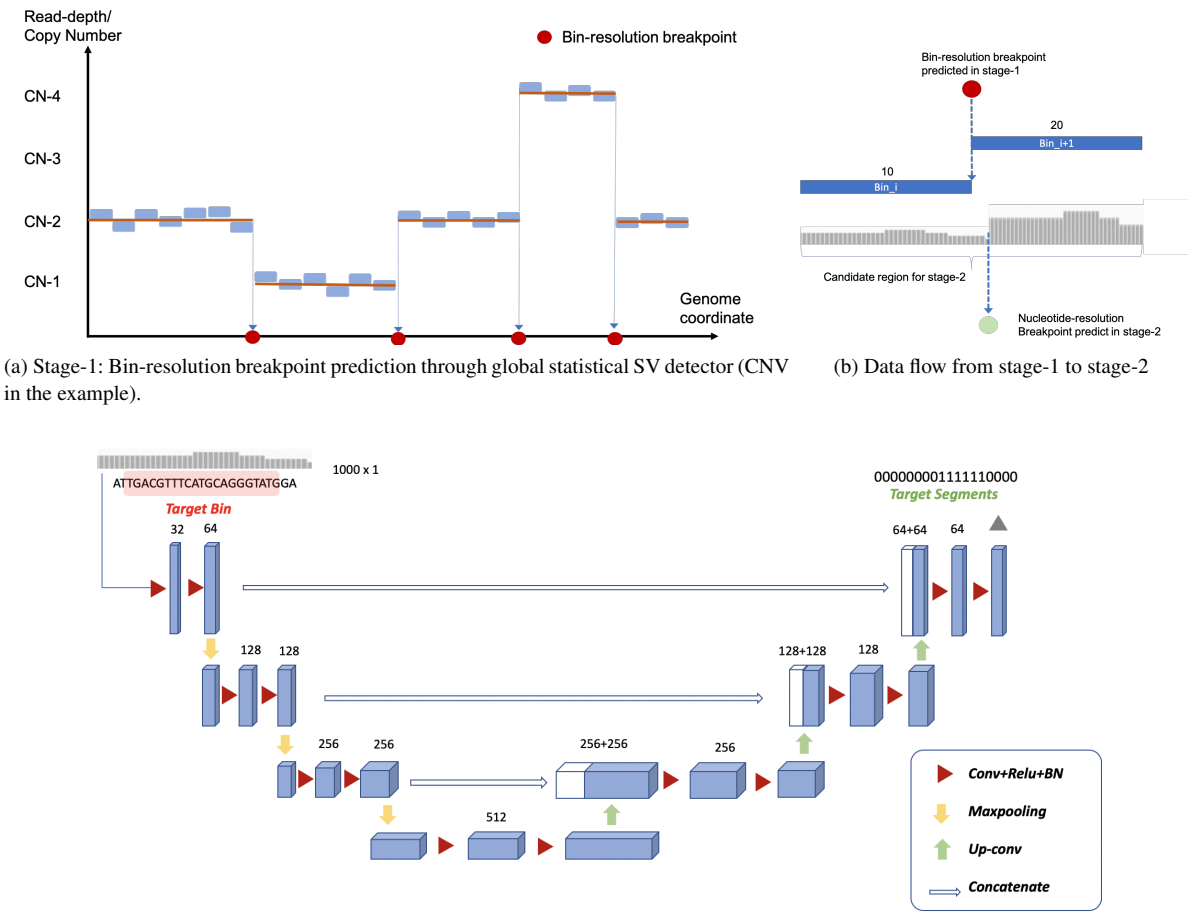
$$DSC(s_{pred}, s_{gold}) = \frac{2\sum_{i=1}^{l} s_{gold}^{i} * s_{pred}^{i} + \epsilon}{\sum_{i=1}^{l} s_{gold}^{i} + \sum_{i=1}^{l} s_{pred}^{i} + \epsilon}$$

Where s_{gold} and s_{pred} are nucleotide label masks $\in \{0, 1\}$, and ϵ is a tolerance needed when there is no 1-masked label for both gold and predict vector. We apply early stopping with minimum 10 epochs and maximum 100 epochs for the training process.

2.3.2 Testing

The key assumption of applying a supervised model for intra-bin SV segmentation task is that *read-depth signals surrounding a breakpoint of a specific SV type share a similar pattern for WGS samples with similar experiment settings (read-depth, especially).* After training a U-net model, we apply the trained model to the same sample or a different sample with similar sequencing setting. For the different sample case, the read-depth signals are standardized separately. Putative regions are generated before

Fig. 1: Nucleotide-resolution SV segmentation pipeline.



(a) Stage-1: Bin-resolution breakpoint prediction through global statistical SV detector (CNV in the example).

(b) Data flow from stage-1 to stage-2

(c) Stage-2: Nucleotide-resolution intra-bin SV segmentation through U-Net model.

applying the trained model. Finally, we get base-pair segmentation of the regions for downstream analysis.

3 Results

3.1 Experiment settings

3.1.1 Data

In this paper, we evaluated the proposed model based on Illumina WGS data from the 1000 Genomes Project (1kGP). NA12878 was used as the benchmark data as previous works. We mainly used two typical sequencing depths, which refer to as 7x and 60x. In addition to the 7x and 60x data, we used 300x WGS data of NA12878 from NIST's Genome in a Bottle (GIAB) Project. For cross-sample evaluation, we used the following samples of both 7x and 60x data: NA19238 and NA19239 from YRI, NA18939 from JPT, HG00419 from CHS.

WGS data were first aligned to a reference genome. BWA-MEM [15] is a choice to align reads to the reference genome of GRCh37 (primary reference including unplaced contigs) as in 1kGP. PCR duplicates were marked for downstream analysis. For those sample with alignment, we directly downloaded the released run-level bam files from 1kGP.

We generated data for the task based on autosomal SV annotation from 1kGP Phase 3 [16]. We parsed the VCF file and selected all SVs containing START and END position information for the above five samples. In total, four types of SV were included: DUP, $DEL(_*)$, and CNV, which represent duplication, deletion, multi-allelic CNV, respectively. The insertions of mobile elements (SVA, $LINE1$ and ALU) were excluded for lacking END position information. SVs with length less than 50 bp were filtered out. For deletion, subtypes of DEL_SVA, DEL_ALU and DEL_LINE1 were included. For each sample, the number of each SV type is summarized in Table 1. For test samples used in the cross-sample evaluation, the number of the NA12878 non-overlapping SVs is also shown in the table.

Table 1. Number of extracted SVs in autosomes of the samples.

Sample	Total SVs	DEL	DUP	CNV	NA12878 non-overlap
NA12878	2041	1889	8	146	0
NA19239	2282	2087	11	184	1137
NA19238	2331	2126	18	187	1140
NA18939	2239	2037	23	179	937
HG00419	2118	1926	14	178	809

We split SVs into training and test set for two experiment settings, separately. For the intra-sample setting, besides the stratified 5-fold cross-validation, SVs were randomly shuffled and split into stratified training (80%) and test set (20%). For the cross-sample setting, SVs from one

sample (NA12878) were used for the training set, while SVs from the other samples were used for test set. We then generated fixed-length bins used for model training and testing based on the SVs. The genome coordinates of each bin were determined following the method described in section 2.3.1. We extracted base-pair read-depth information for each bin X with *pysam* (Version 0.15.0). Low base quality (Phred < 30) ones were ignored in the depth counting. For those bins with all-zero values (no signal at all), we filtered them out. For each sample, bin vectors X were standardized using the mean and standard derivation of its read-depth. We split whole-genome into non-overlapping bins with the pre-fixed bin-size and filtered out low mappability (mappability score < 0.9) bins based on the mappability file from ENCODE project. We then randomly sampled 1% of the bins for calculating the mean and standard derivation.

3.1.2 Evaluation metric

For the segmentation task, we evaluated models using Dice similarity coefficient for all bins (DSC-ALL) and SV-containing bins (DSC-BK). For the breakpoints of SV in 1kGP, the precision of each one is different. Some of them do not contain confidence interval (CI) information (CI-contained:32182 SVs, no CI-contained:18828 SVs). Therefore, instead of evaluating the position of those breakpoints exactly, we used Dice similarity coefficient as an approximation. Up to the point of this work, there was few work focusing on predicting SV segments using base-pair RD information for a single sample. To compare model performances with the SVM model used in the SVclassify work [17], we generated SV-containing and non-SV-containing labels based on the segmentation results for a simplified classification task. The classification performance was evaluated using AUC, sensitivity and false discovery rate (FDR).

3.1.3 Model parameters

Besides the U-net model (UNet), we applied a classic two-layer convolutional neural network (CNN) for the segmentation task. For training those deep learning models, we split 20% of the training data as the development set and used hyperOpt[18] to automatically search pre-defined hyper-parameter space. More details of each model are described in the supplement document. The default settings used in the following experiments are: UNet_param={conv_window=7, batch_size=64, learning_rate=0.001, maxpooling_len=[5,5,2,2,5,5], BatchNorm=True, dropout=0.2 }, CNN_param={conv_window=7, batch_size=64, learning_rate=0.001, kernel_size=[64,128], maxpooling_len=[2,2], BatchNorm=False, dropout=0.2, fc_hidden_num=256 }. In total, UNet has 6M parameters, while CNN has 16M parameters.

3.2 Performance on NA12878 benchmark data

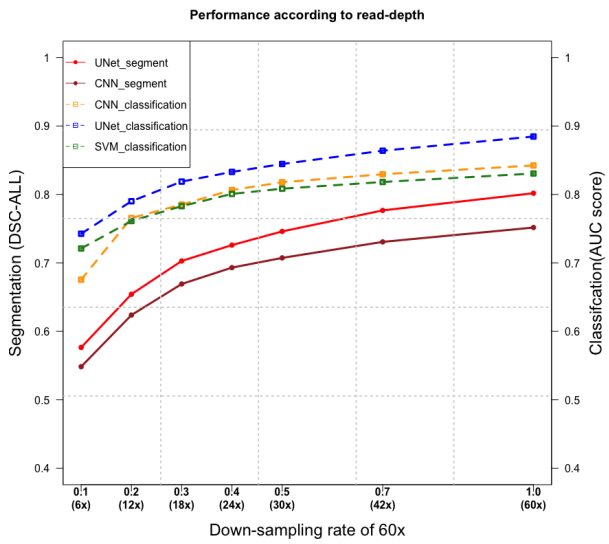
Table 2. 5-fold cross-validation results on NA12878 with 1000 bp bin-size. For UNet and CNN trained on GPU, we repeated fives times for the same setting and calculated the average results. The best performance for each depth is highlighted with underline.

Depth	Model	Classification			Segmentation	
		AUC	Sensitivity	FDR	DSC-ALL	DSC-BK
7x	SVM	0.7609	0.7307	0.2218	/	
	CNN	0.7552	0.7549	0.2434	0.6031	0.6577
	UNet	0.7742	0.7425	0.2052	0.6342	0.6918
60x	SVM	0.8306	0.8025	0.1496	/	
	CNN	0.8424	0.8187	0.1402	0.7517	0.7884
	UNet	0.8846	0.8599	0.0949	0.8018	0.8322
300x	SVM	0.8783	0.8812	0.1237	/	
	CNN	0.8955	0.8910	0.1004	0.7951	0.8327
	UNet	0.9189	0.9127	0.0756	0.8313	0.8651

We first performed 5-fold cross-validation on the benchmark data of NA12878 for single sample evaluation with a bin-size of 1000 bp. The classification and segmentation results are shown in Table 2. For all the models, performance improves with the increase of read-depth. In the binary classification task, there are 9.1%, 11.5% and 14.3% relative increase (with statistical significance) on AUC from 7x to 60x for SVM, CNN and UNet, respectively. From 60x to 300x, there are 5.7%, 6.3% and 3.9% relative increase with statistical significance over AUC. For each model, the sensitivity increases and the FDR reduces with the increase of read-depth. In the segmentation task, the improvement is more significant from 7x to 60x, due to the lower performance on 7x data. There are 24.6% and 26.4% relative improvement over DSC-ALL score for CNN and UNet. From 60x to 300x, the gains reduce to 5.8% and 3.7% for CNN and UNet. On all the three depth data, UNet performs better the other two models. On 7x data, the classification performances are relatively close. UNet achieves a slightly higher AUC with lowest FDR. On 60x and 300x data, UNet performs significantly better with both higher sensitivity and lower FDR. When compared with CNN in segmentation task, DSC-ALL score of UNet is 6.7% and 4.6% higher on 60x and 300x data.

3.2.1 Read-depth effect

Fig. 2: Performance of different read-depths through down-sampling 60x WGS of NA12878. The dashed curves connect AUC scores of classification, while the line curves show the DSC-ALL scores of segmentation.

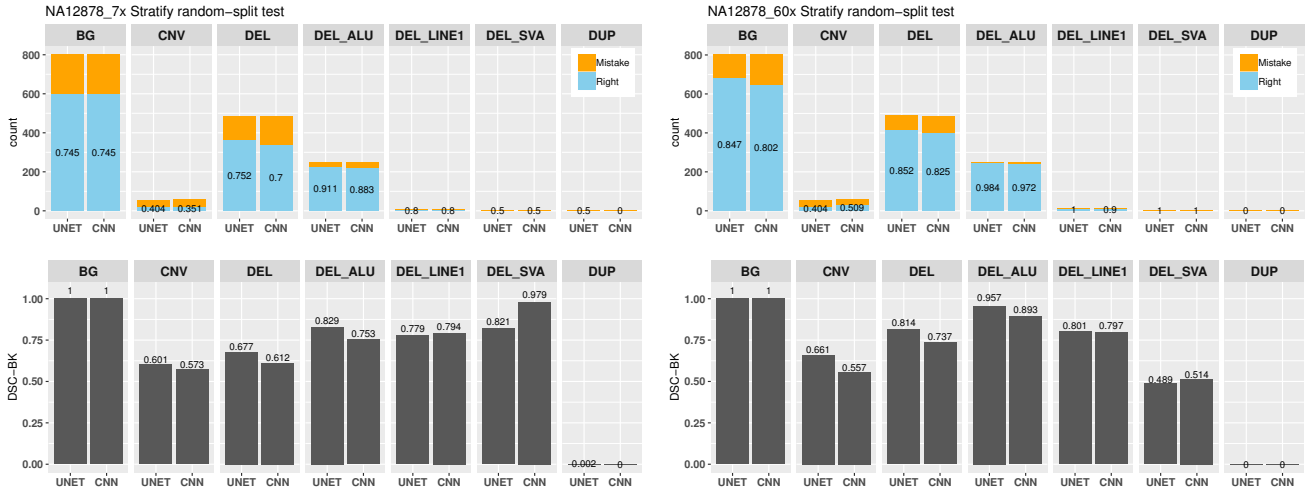


We empirically evaluated how read-depth affects model performance in a more refinement step through down-sampling the NA12878 60x WGS data. We conducted 5-fold cross-validation for the WGS data of different down-sampling rate. In Figure 2, we can observe a general trend of performance improvement as the increase of read-depth. Curves in the region of down-sampling rate below 0.5 are relatively more steep when compared with curves in the rest region. It indicates the difficulty of doing intra-bin SV detection for a single sample in nucleotide-resolution using low-depth data. Based on performance gaps shown in the figure, 30x and 40x are empirically suggested minimal read-depth used for for intra-bin SV classification and segmentation for a single sample.

Table 3. 5-fold cross-validation results of 100 bp, 500 bp and 1000 bp bin-size on 60x data. For UNet and CNN trained on GPU, we repeated fives times for the same setting and calculated the average results. The underline indicates the best score for each model on the three bin-size and bold font indicates the best performance among all models on all the three bin-sizes.

Bin size	Classification									Segmentation			
	SVM			CNN			UNet			CNN		UNet	
	AUC	Sensitivity	FDR	AUC	Sensitivity	FDR	AUC	Sensitivity	FDR	DSC-ALL	DSC-BK	DSC-ALL	DSC-BK
100 bp	0.8343	0.8215	0.1588	0.8445	0.8362	0.1510	0.8502	0.8316	0.1373	0.7235	0.7682	0.7383	0.7829
500 bp	0.8335	0.8214	0.1583	0.8518	0.8476	0.1448	0.8756	**0.8750**	0.1237	0.7586	0.7990	0.7928	**0.8325**
1000 bp	0.8306	0.8025	0.1496	0.8424	0.8187	0.1402	**0.8846**	0.8599	**0.0949**	0.7517	0.7884	**0.8018**	0.8322

Fig. 3: Classification and segmentation performance of CNN and UNet for each SV type. Figures in the first row show the number of correctly classified bins and their proportion. Figures in the second row show the dice coefficient score for the correctly classified bins.



3.2.2 Bin size effect

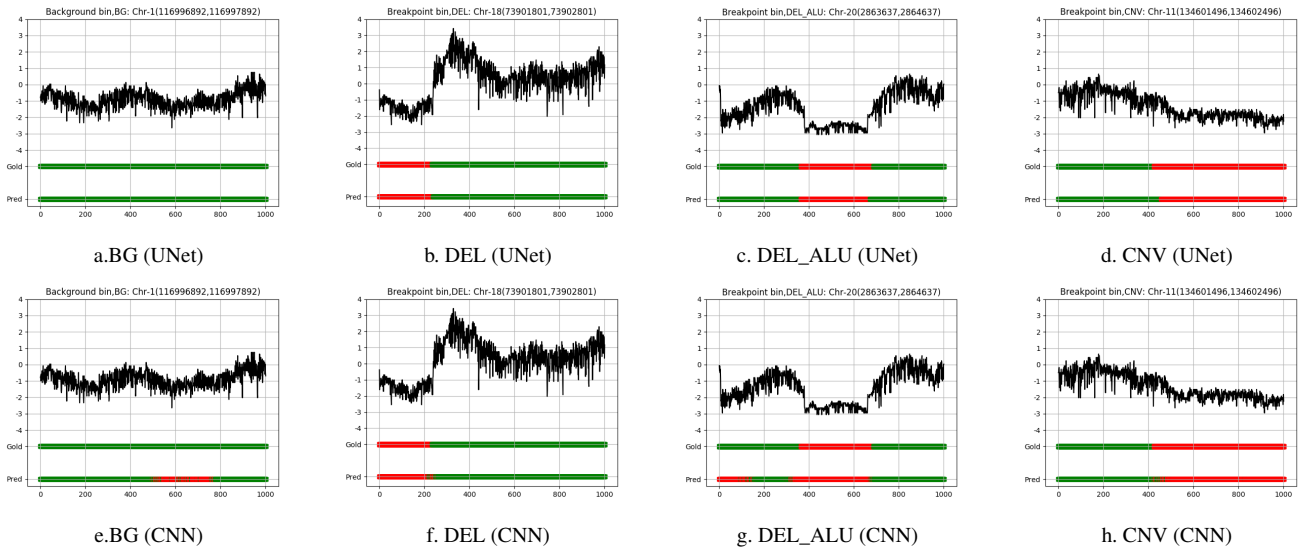
We examined model performances with three different bin-sizes: 100 bp, 500 bp and 1000 bp for the WGS data of 60x read-depth. In the binary classification task, the AUCs do not show significant differences among the three bin-size settings for SVM and CNN. For SVM, the AUC slightly decreases as the increase of bin-size from 100 bp to 500 bp. CNN gets its best AUC of 0.8518 with 500 bp bin-size. For UNet, AUC is improved as the increase of bin-size. The best classification result achieved by UNet is 0.8846 at the bin size of 1000 bp. For all the three models, FDR reduces as the increase of bin-size. In the segmentation task, we can observe a significant performance gap between 100 bp and 500 bp (round 4.9% and 7.4% improvement in DSC-ALL from 100 bp to 500 bp for CNN and UNet, respectively), while the performances are close but different with statistical significance between 500 bp and 1000 bp.

From a data perspective, as the increase of the bin-size, more context RD signals are considered. This is a benefit for reducing FDR of non-SV-containing bins, while it also has an effect of sensitivity reduction for SV-containing bins. From a model perspective, SVM is a shallow model with equivalent one-flatten-layer. CNN and U-net are deep neural networks consist of hierarchical multiple layers and shared local filters for the input layer. The local shared structure makes them less sensitive to the change of bin size in lower layers when compared with SVM. But the number of hidden units in a higher level reduces significantly as bin size decreases. This could be a reason that the performances of deep learning models decrease significantly when the bin-size is less than 500 bp. This effect is more obvious for UNet, which has more layers. In general, for each of the three bin-size, the deep neural network models (CNN and UNet) perform better than SVM, and UNet achieves the best performance in both classification and segmentation task.

3.2.3 Intra-bin segmentation result analysis

In this section, we investigate the intra-bin segmentation performance in more detail. We show the classification and segmentation result for each SV type in Figure 3 (results on 300x data is shown in the supplement document). As the test set is stratified, the distribution of SV types in the figure also reflects the overall SV distribution on the whole dataset. For SV-overlapping bins, the major SV type is the deletion, which holds around 92%. The rest SV types are CNV (7%) and DUP (0.2%) For deletion, the proportion of four subtypes in all SVs are: DEL(60%), DEL_ALU(31%), DEL_LINE1(1%) and DEL_SVA(0.2%). Under such data configuration, the trained intra-bin segmentation models mainly learn features associated with the dominant-label of deletion. Consistent with the cross-validation results, the performance improvement brought by higher RD can be also observed for major labels (number >=50) of BG, DEL and DEL_ALU in both CNN and UNet. For non-sv-overlapping bins of BG, more correct predictions are made by UNet than CNN on 60x data. For major sv-overlapping bins of DEL and DEL_ALU, UNet not only classifies more bins correctly (first row of Figure 3), but also gives better segmentation results for those bins on both 7x and 60x (second row of Figure 3), when compared with CNN. For the minor sv-overlapping bins of CNV, DUP, DEL_SVA and DEL_LINE1, the performances are more variant due to the small number of sample. From the figure, UNet performs slightly better than CNN on the segmentation of CNV bins on both 7x and 60x data. CNN

Fig. 4: Examples of the intra-bin segmentation results of four major types (BG, DEL, DEL_ALU and CNV). Red color represents SV region, while green color represents background region. Gold standard segmentation is shown in the first line below RD signals, while prediction is shown in the second line.



a.BG (UNet) b. DEL (UNet) c. DEL_ALU (UNet) d. CNV (UNet)

e.BG (CNN) f. DEL (CNN) g. DEL_ALU (CNN) h. CNV (CNN)

performs slightly better than UNet for DEL_SVA and DEL_LINE1 on 7x data, and the gap further reduces on 60x data.

Both UNet and CNN achieve good performances on DEL_ALU segmentation, which is even better than their performance on DEL bins. On 60x data, UNet correctly detects 98.4% DEL_ALU bins in a total of 248 DEL_ALU bins and achieves 0.957 DSC-BK score. CNN detects 97.1% DEL_ALU bins, but has a lower segmentation performance of 0.893 DSC-BK score.

We demonstrate the examples of the intra-bin segmentation in Figure . We chose examples from the major bin labels of BG, DEL, DEL_ALU and CNV. CNV and UNet segmentation results are given for the selected bin regions. In general, based on the segmentation results on the test data, the UNet shows two advantages. **First, UNet predicts more integrated segments.** For example, in Figure (e,f,g,h), CNN makes more fragmentary segments, especially around breakpoint boundaries, while the segments given by UNet are integrated (Figure (b,c,d)). **Second, UNet has lower false discovery fragment.** As shown in Table 2, UNet achieves the lowest FDR. In the Figure (a), for the bin region of Chr18:(73901801, 73902801), the RD signal is even difficult for human to make a judgment. The UNet model successfully avoids any segmentation on the region, while generates closely correct segmentation for a "not so obvious" signals in the bin of Chr11:(134601496, 134602498) (Figure (d)). CNN makes a false positive segment in the region of Chr18:(73901801, 73902801).

3.2.4 data augmentation

We conducted data augmentation to increase the amount of the training data. For each SV breakpoint, we took a different random shift to get a new start position deviating from the SV breakpoints. To keep the data balance, we generated the same amount of bins without overlapping any SV breakpoints through random sampling. Each augmentation epoch generates the same amount of data as the original dataset. We did 10 epochs on the training data and evaluated on the non-augmented test set. The result is summarized in Table 4.

In general, for all the three models, FDR decreases after the data augmentation. But sensitivity does not necessarily increase. On 7x data, the performance of data-augmented UNet even decreases for the significant decrease of the sensitivity. Because the SV annotation used here is not perfect and 7x data is noisy for its low coverage. Although the data augmentation has a general positive effect on overall performance for SVM and CNN, UNet without data augmentation still performs better than SVM and UNet with the data augmentation.

3.3 Cross-sample evaluation

To investigate whether the segmentation model can be generalized across different samples, we performed a cross-sample evaluation. We trained on the whole dataset of a sample and tested on the non-overlapping regions on the other. In this experiment, the benchmark data NA12878 was used as the training set, while the other four samples were used for testing. The results on 60x depth WGS are shown in Table 5. Note that 60x depth is not the exact average depth for every sample, it is a rough indicator for samples with similar sequencing depth. For the three samples of NA19239, NA18939, and HG00419, UNet achieves the highest score on AUC and DSC(-ALL/-BK). For NA19238, CNN performed slightly better than UNet (around 0.009 and 0.003 absolute improvement on AUC and DSC-ALL, respectively), but it is not statistically significant (p-value$_{AUC}$=0.13, p-value$_{DSC_ALL}$=0.77). In all the four test samples, UNet has the lowest FDR. The two deep models, CNN and UNet, perform better than SVM in the classification task on the dataset. From the results, we can observe that UNet keeps its low FDR character in the cross-sample evaluation, which is important for detecting recurrent SVs across different samples.

4 Discussion

In this paper, we focused on the challenging task of segmenting base-pair RD signals inside of a bin. We proposed a novel approach of using U-net model to directly process noisy base-pair RD signals, which was impractical for directly applying previous statistical RD-based approaches. The U-net model uses an Encoder-and-decoder network structure and generates segmentation on original sequences. To the best of our knowledge, this is the first attempt to apply the U-net model for genome data. We conducted a set of experiments on real Illumina WGS

Table 4. 10-epochs data augmentation results on NA12878 stratified test data. For UNet and CNN trained on GPU, we repeated fives times for the same setting and calculated the average results.

Depth	DA	Classification									Segmentation			
		SVM			CNN			UNet			CNN		UNet	
		AUC	Sensitivity	FDR	AUC	Sensitivity	FDR	AUC	Sensitivity	FDR	DSC-ALL	DSC-BK	DSC-ALL	DSC-BK
7x	None	0.7447	0.7246	0.2445	0.7387	0.7630	0.2656	**0.7641**	0.7692	0.2370	0.6058	0.6662	**0.6368**	0.7113
	da-10	0.7559	0.7357	0.2329	0.7541	0.7493	0.2391	0.7406	0.5747	0.1396	0.6326	0.6880	0.5462	0.5669
	improved	+	+	+	+	-	+	-	-	+	+	+	-	-
60x	None	0.8126	0.7903	0.1727	0.8273	0.8283	0.1728	0.8681	0.8323	0.1026	0.7345	0.786	**0.7891**	0.8276
	da-10	0.8175	0.7841	0.1596	0.8433	0.8320	0.1479	**0.8767**	0.8362	0.0882	0.7728	0.8189	0.7880	0.8233
	improved	+	-	+	+	+	+	+	+	+	+	+	-	-
300x	None	0.8638	0.8831	0.1497	0.8771	0.8816	0.1261	0.9097	0.9152	0.0947	0.8000	0.8399	0.8296	0.8666
	da-10	0.8762	0.8794	0.1261	0.8963	0.9142	0.1173	**0.9226**	0.9005	0.0578	0.8250	0.8616	**0.8494**	0.8722
	improved	+	-	+	+	+	+	+	-	+	+	+	+	+

Table 5. Cross-sample evaluation on 60x WGS data with 1000 bp bin-size. For UNet and CNN trained on GPU, we repeated fives times for the same setting and calculated the average results. Bold font is the best score for each test sample. (†) indicates improvement with statistical significance (t-test, $P < 0.01$).

Sample	Classification									Segmentation			
	SVM			CNN			UNet			CNN		UNet	
	AUC	Sensitivity	FDR	AUC	Sensitivity	FDR	AUC	Sensitivity	FDR	DSC-ALL	DSC-BK	DSC-ALL	DSC-BK
NA19239	0.7324	0.5896	0.1741	0.7420	0.5675	0.1277	**0.7770** †	**0.6166**	**0.0918**	0.5571	0.5811	**0.6466** †	0.6702
NA19238	0.7213	0.6275	0.2276	**0.7713**	**0.6350**	0.1259	0.7622	0.5639	**0.0660**	**0.6227**	0.6533	0.6199	0.6373
NA18939	0.7116	0.6363	0.2498	0.7268	0.7032	0.2615	**0.7618**†	**0.7220**	**0.2049**	0.5624	0.6342	**0.6025** †	0.6756
HG00419	0.7308	0.6907	0.2480	0.7501	0.7573	0.2512	**0.7967** †	**0.8004**	**0.2040**	0.6158	0.6916	**0.6670** †	0.7423

data. In both segmentation and simplified classification task, the U-net model generally achieves better performances when compared with CNN and SVM. We also empirically evaluated the effect of read-depth and bin-size for different model performances. In the cross-sample evaluation, the experiment shows the generalization of the U-net model across different samples with a similar read-depth.

The proposed method can be used in the following two scenarios. First, searching for specific SVs inside of a bin for a single sample. Given a set of target SV annotations with base-pair RD signals, we can train a UNet model and search similar RD signatures associated with the target SVs on a whole-genome scale. Second, cross sample SV detection, which is used as an enhancement to traditional bin-based approaches. For the recent work of DeepVariant, it applied CNN on pileup images to detect SNP and small-indel variants in putative regions (the bin-size is usually set to 100 bp). This work is different from the deepVariant in the following two aspects. First, we focused on SVs that are larger than 50 bp and use one-dimensional base-pair RD signals as the input. Second, for SV detection, we formulated the problem as a segmentation task instead of a classification task. Therefore, the proposed U-net model can not only predict whether a bin contains any SV, but also can make SV segmentation for the bin in nucleotide-resolution.

The proposed U-net model can be also applied to the RD signals generated by long-read sequencing, such as the single molecule real-time (SMRT) sequencing technology from Pacific Biosciences (PacBio). When compared with the short-sequencing data, the depth-of-coverage signals in long-sequencing data is relatively more smooth than that of short-sequencing data. Therefore, we expect the long-read sequence would be easier for the U-net model to do the intra-bin segmentation. In this work, we only used RD signals to detect intra-bin SV segments. For the reason that the SV annotation is assembled from multiple sources of information, for some cases, using RD information only is not possible to make exact predictions. As an automatic feature learning framework, it is more flexible to incorporate other sources of biological signals in the U-net. We intend to explore other SV associated biological signatures besides the RD signals in the framework of U-net. Also, we can also apply the U-net framework to other similar tasks, such as transcript boundary detection [19].

References

[1]Tobias Rausch, David TW Jones, Marc Zapatka, Adrian M Stütz, Thomas Zichner, Joachim Weischenfeldt, Natalie Jäger, Marc Remke, David Shih, Paul A Northcott, et al. Genome sequencing of pediatric medulloblastoma links catastrophic dna rearrangements with tp53 mutations. *Cell*, 148(1):59–71, 2012.

[2]Mario Falchi, Julia Sarah El-Sayed Moustafa, Petros Takousis, Francesco Pesce, Amélie Bonnefond, Johanna C Andersson-Assarsson, Peter H Sudmant, Rajkumar Dorajoo, Mashael Nedham Al-Shafai, Leonardo Bottolo, et al. Low copy number of the salivary amylase gene predisposes to obesity. *Nature genetics*, 46(5):492, 2014.

[3]BV Hooli, Zs M Kovacs-Vajna, K Mullin, MA Blumenthal, Manuel Mattheisen, C Zhang, C Lange, G Mohapatra, Lars Bertram, and RE Tanzi. Rare autosomal copy number variations in early-onset familial alzheimerâŁ™s disease. *Molecular psychiatry*, 19(6):676, 2014.

[4]Babak Alaei-Mahabadi, Joydeep Bhadury, Joakim W Karlsson, Jonas A Nilsson, and Erik Larsson. Global analysis of somatic structural genomic alterations and their impact on gene expression in diverse human cancers. *Proceedings of the National Academy of Sciences*, 113(48):13768–13773, 2016.

[5]Alexej Abyzov, Alexander Eckehart Urban, Michael Snyder, and Mark Gerstein. Cnvnator: An approach to discover, genotype and characterize typical and atypical cnvs from family and population genome sequencing. *Genome research*, pages gr–114876, 2011.

[6]Tobias Rausch, Thomas Zichner, Andreas Schlattl, Adrian M Stütz, Vladimir Benes, and Jan O Korbel. Delly: structural variant discovery by integrated paired-end and split-read analysis. *Bioinformatics*, 28(18):i333–i339, 2012.

[7]Ruping Sun, Michael I Love, Tomasz Zemojtel, Anne-Katrin Emde, Ho-Ryun Chung, Martin Vingron, and Stefan A Haas. Breakpointer: using local mapping artifacts to support sequence breakpoint discovery from single-end reads. *Bioinformatics*, 28(7):1024–1025, 2012.

[8]Zamin Iqbal, Mario Caccamo, Isaac Turner, Paul Flicek, and Gil McVean. De novo assembly and genotyping of variants using colored de bruijn graphs. *Nature genetics*, 44(2):226, 2012.

[9]Babak Alipanahi, Andrew Delong, Matthew T Weirauch, and Brendan J Frey. Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. *Nature biotechnology*, 33(8):831, 2015.

[10]Jian Zhou and Olga G Troyanskaya. Predicting effects of noncoding variants with deep learning–based sequence model. *Nature methods*, 12(10):931, 2015.

[11]Ryan Poplin, Pi-Chuan Chang, David Alexander, Scott Schwartz, Thomas Colthurst, Alexander Ku, Dan Newburger, Jojo Dijamco, Nam Nguyen, Pegah T Afshar, et al. A universal snp and small-indel variant caller using deep neural networks. *Nature biotechnology*, 36(10):983, 2018.

[12]Hamid Eghbal-zadeh, Lukas Fischer, Niko Popitsch, Florian Kromp, Sabine Taschner-Mandl, Khaled Koutini, Teresa Gerber, Eva Bozsaky, Peter F Ambros,

Inge M Ambros, et al. Deep snp: An end-to-end deep neural network with attention-based localization for break-point detection in snp array genomic data. *arXiv preprint arXiv:1806.08840*, 2018.

[13]Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[14]Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[15]Heng Li. Aligning sequence reads, clone sequences and assembly contigs with bwa-mem. *arXiv preprint arXiv:1303.3997*, 2013.

[16]Peter H Sudmant, Tobias Rausch, Eugene J Gardner, Robert E Handsaker, Alexej Abyzov, John Huddleston, Yan Zhang, Kai Ye, Goo Jun, Markus Hsi-Yang Fritz, et al. An integrated map of structural variation in 2,504 human genomes. *Nature*, 526(7571):75, 2015.

[17]Hemang Parikh, Marghoob Mohiyuddin, Hugo YK Lam, Hariharan Iyer, Desu Chen, Mark Pratt, Gabor Bartha, Noah Spies, Wolfgang Losert, Justin M Zook, et al. svclassify: a method to establish benchmark structural variant calls. *BMC genomics*, 17(1):64, 2016.

[18]James Bergstra, Dan Yamins, and David D Cox. Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms. In *Proceedings of the 12th Python in Science Conference*, pages 13–20. Citeseer, 2013.

[19]Mingfu Shao, Jianzhu Ma, and Sheng Wang. Deepbound: accurate identification of transcript boundaries via deep convolutional neural fields. *Bioinformatics*, 33(14):i267–i273, 2017.