



Version 1

Chromosome-scale genome assembly of kiwifruit *Actinidia eriantha* with single-molecule sequencing and chromatin interaction mapping V.1



DOI

<https://dx.doi.org/10.17504/protocols.io.vfue3nw>

Xuepeng Sun¹, Yongsheng Liu², Zhangjun Fei¹

¹Boyce Thompson Institute, Ithaca, NY 14853, USA;

²School of Horticulture, Anhui Agricultural University, Hefei 230036, China



Xuepeng Sun

Create & collaborate more with a free account

Edit and publish protocols, collaborate in communities, share insights through comments, and track progress with run records.

Create free account



DOI: <https://dx.doi.org/10.17504/protocols.io.vfue3nw>

Protocol Citation: Xuepeng Sun, Yongsheng Liu, Zhangjun Fei . Chromosome-scale genome assembly of kiwifruit *Actinidia eriantha* with single-molecule sequencing and chromatin interaction mapping. **protocols.io**

<https://dx.doi.org/10.17504/protocols.io.vfue3nw>

Manuscript citation:

License: This is an open access protocol distributed under the terms of the **Creative Commons Attribution License**, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited

Created: November 08, 2018



Last Modified: November 09, 2018

Protocol Integer ID: 17620

Keywords: Genome assembly, Hi-C, PacBio, Gene prediction, Plant genome, Single molecular sequencing, scale genome assembly of kiwifruit actinidia eriantha, annotation of kiwifruit actinidia eriantha genome, kiwifruit actinidia eriantha genome, scale genome assembly, kiwifruit actinidia eriantha, chromatin interaction mapping this protocol, chromatin interaction mapping, chromosome

Abstract

This protocol includes a computational pipeline used in assembly and annotation of Kiwifruit Actinidia eriantha genome.



Illumina raw reads cleaning

- 1) Deduplication with super_deduper (<https://github.com/dstreett/Super-Deduper>)

Command

```
super_deduper -s 5 -l 40 -p prefix -1 read1.fq -2 read2.fq
```

- 2) Adaptor trimming with trim_galore (<https://github.com/FelixKrueger/TrimGalore>)

Command

```
trim_galore --phred33 --illumina --paired read1.fq read2.fq
```

- 3) Mate-pair reads cleaning with nextclip (<https://github.com/richardmleggett/nextclip>)

Command

```
nextclip --remove_duplicates --min_length 20 --number_of_reads 60 --  
log log.txt --number_of_reads 10000000 -i read1.fq -j read2.fq -o out
```



PacBio assembly

2 Assembly with Canu (<https://github.com/marbl/canu>)

Command

```
canu -p prefix corOutCoverage=50 -d Spinach-pacbio_all
genomeSize=705m -pacbio-raw Pacbio.fastq.gz >>canu.log
```

Command

```
pbalign --tmpDir tmp --nproc 6 --concordant --hitPolicy randombest --
minAccuracy 70 --minLength 50 --algorithmOptions "--minMatch 12 --
bestn 10 --minPctIdentity 70.0\" subreads.bam contigs.fasta
aligned.subreads.bam >aligned.log

variantCaller -j 64 --algorithm arrow -r contigs.fasta --coverage 150
--diploid --minConfidence 40 --minCoverage 5 -o variants.gff -o
canu.consensus.fasta -o canu.consensus.fastq aligned.subreads.rh.bam
>variantCaller.log
```

Assembly with wtdbg (<https://github.com/ruanjue/wtdbg>)

**Command**

```
wtdbg-1.1.006 -t 96 -i pb-reads.fa -o dbg -H -k 21 -S 1.02 -e 3 2>&1  
| tee log.wtdbg
```

```
wtdbg-cns -t 96 -i dbg.ctg.lay -o dbg.ctg.lay.fa -k 15 2>&1 | tee  
log.cns.1
```

Assembly merge with Quickmerge (<https://github.com/mahulchak/quickmerge>)

Command

```
merge_wrapper.py assembly1.fasta assembly2.fasta
```

Polish of PacBio assembly with Illumina reads

3



Command

```
bowtie2 --rf --no-unal -I min -X max -x index -1 read1.fq -2 read2.fq  
-S align.sam
```

```
java -Xmx300G -jar pilon-1.22.jar --genome contig.fa --frags PE.bam --  
jumps MP.bam --output out --outdir outdir --changes --vcf --tracks --  
diploid --fix snps,indels --threads 60 --flank 0
```

Scaffold anchoring with Hi-C data

4 Anchoring with LACHESIS (<https://github.com/shendurelab/LACHESIS>)

Using recommended protocol with parameters to be set as

```
"CLUSTER_MIN_RE_SITES=48, CLUSTER_MAX_LINK_DENSITY=2,  
CLUSTER_NONINFORMATIVE_RATIO=2, ORDER_MIN_N_RES_IN_TRUN=14,  
ORDER_MIN_N_RES_IN_SHREDS=15"
```

Repeat annotation

5 Identification of MITE

Command

```
perl MITE_Hunter_manager.pl -i assembly.fasta -n 64 -c 64 -S 12345678  
cat genome_Step8_*.fa genome_Step8_singlet.fa > MITE.lib
```

Identification of LTR

Details can be found in Campbell MS, Law M, Holt C, Stein JC, Moghe GD, Hufnagel DE, Lei J, Achawanantakun R, Jiao D, Lawrence CJ. 2014. MAKER-P: a tool kit for the rapid



creation, management, and quality control of plant genome annotations. Plant physiology 164:513-524.

Masking genome with MITE and LTR libraries

Command

```
RepeatMasker -pa 64 -lib MITE_LTR.lib -dir . assembly.fasta
```

Identification of novel repeats with RepeatModeler

Command

```
perl rmaskedpart.pl assembly.fasta.masked 50 > umseqfile
```

```
BuildDatabase -name umseqfiledb -engine ncbi umseqfile
```

```
RepeatModeler -pa 64 -database umseqfiledb
```

Gene prediction

6 Preparation of transcriptome evidence

Command

```
#de novo assembly
Trinity --seqType fq --max_memory 200G --CPU 50 --normalize_reads --
left r1.fq --right r2.fq --output trinity_denovo

#genome guided assembly
STAR --genomeDir gd --alignIntronMax 80000 --twopassMode Basic --
runThreadN 50 --readFilesCommand zcat --readFilesIn r1.fq r2.fq --
outFileNamePrefix mapped_star
Trinity --genome_guided_bam mapped_starAligned.out.s.bam --max_memory
200G --genome_guided_max_intron 80000 --CPU 50 --output trinity_GG

cat Trinity-DN.fasta Trinity-GG.fasta > transcripts.fasta

#stringtie assembly
hisat2 --dta -p 64 -x assembly.fasta -1 r1.fq -2 r2.fq -S
hisat2.map.sam

stringtie hisat2.map.sort.bam -o stringtie.gtf -p 60

$PASA_HOME/misc_utilities/accession_extractor.pl < Trinity-DN.fasta >
tdn.accs

$PASA_HOME/scripts/Launch_PASA_pipeline.pl \
-c $PASA_HOME/pasa_conf/alignAssembly.config \
--MAX_INTRON_LENGTH 60000 \
--cufflinks_gtf stringtie.gtf \
-C -R --CPU 64 \
-g aseembly.fa \
-t transcripts.fasta \
--TDN tdn.accs \
--ALIGNERS blat,gmap
```

Model training with Braker



Command

```
braker.pl \  
--cores=64 \  
--BAMTOOLS_PATH=bamtools-2.4.1-0/bin \  
--AUGUSTUS_BIN_PATH=/bin \  
--AUGUSTUS_CONFIG_PATH=config \  
--AUGUSTUS_SCRIPTS_PATH=/bin \  
--genome=assembly.fa \  
--species=species \  
--softmasking=1 \  
--prot_seq=homologous_protein.fa \  
--prg=spaln \  
--ALIGNMENT_TOOL_PATH=spaln/bin \  
--bam=rnaseq.sort.bam
```

Gene prediction and intergration with MAKER-P

Following protocol in Campbell MS, Law M, Holt C, Stein JC, Moghe GD, Hufnagel DE, Lei J, Achawanantakun R, Jiao D, Lawrence CJ. 2014. MAKER-P: a tool kit for the rapid creation, management, and quality control of plant genome annotations. Plant physiology 164:513-524.

BUSCO evaluation

7

Command

```
python run_BUSCO.py -i protein.fa -o pep_busco -l embryophyta_odb9/ -  
m proteins -c 60
```



K-mer analysis

8

Command

```
jellyfish count -C -m 17 -s 100000000000 -t 40 -o db.jf *.fq  
jellyfish histo -t 40 db.jf > db.histo
```

Molecular dating and gene family evolution

9 Orthogroup construction

Command

```
orthofinder -t 64 -M msa -A mafft -I 1.5 -f dir
```

Phylogeny with single-copy orthogroups

Command

```
for i in OG*.fa;
do
mafft --maxiterate 1000 --localpair --thread 60 $i > ${i}.mafft;

trimal -automated1 -in ${i}.mafft -out ${i}.mafft.trimal;
done
```

Concatenate alignments

Maximum likelihood phylogeny construction

Command

```
iqtree -s alignment.phy -nt AUTO
```

Molecular dating with MCMCTree

The protocol is described in the manual

(<http://abacus.gene.ucl.ac.uk/software/MCMCtree.Tutorials.pdf>). We used "Relaxed rate" and two time constraints: 5-10 mya for tomato and potato, < 5 mya for two kiwifruits. The maximum time limit for the root is 150 mya.

Gene family evolution with CAFE

Command

```
# pull out large gene families
python
~/tools/CAFE/python_scripts/cafetutorial_clade_and_size_filter.py -i
Orthogroups.GeneCount.reformat -o largeFam -s

cafe
tree tree_estimated_from_MCMCTree;
load -i filtered_cafe_input.txt -filter -l
filtered_cafe_input.logfile -p 0.05
lambda -s
report filtered_cafe_input.cafe

#for large families, assign a lambda value estimated above with
command "lambda -l"
```