Ensembl Gene Annotation (e!107)

Xenopus tropicalis (GCA_000004195.4)

Table of Contents

This document describes the annotation process of an assembly. The first stage is Assembly Loading where databases are prepared and the assembly loaded into the database.

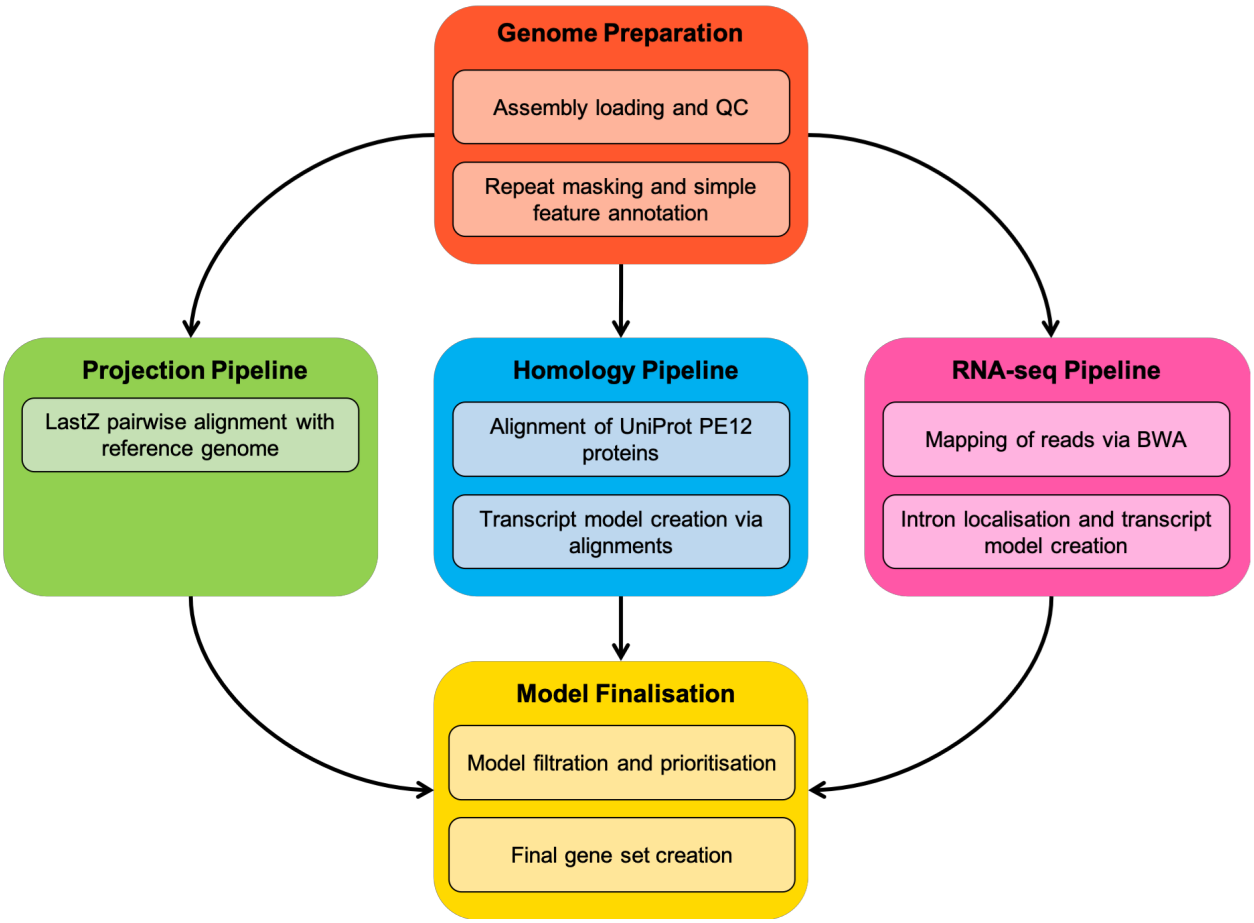

Fig. 1: Flowchart of the protein-coding annotation pipeline. Small ncRNAs, Ig genes, TR genes, and pseudogenes are computed using separate pipelines.

Section 1: Genome Preparation

The genome phase of the Ensembl gene annotation pipeline involves loading an assembly into the Ensembl core database schema and then running a series of analyses on the loaded assembly to identify an initial set of genomic features.
The most important aspect of this phase is identifying repeat features (primarily through RepeatMasker) as soft masking of the genome is used extensively later in the annotation process.

Repeat Finding

After the genomic sequence has been loaded into a database, it is screened for sequence patterns including repeats using RepeatMasker [1] (version 4.0.5 with parameters, *-nolow -engine "crossmatch"*), Dust [2] and TRF [3].
For the *Xenopus tropicalis* annotation, the Repbase vertebrate library was used with RepeatMasker. In addition to the Repbase library, where available, a custom repeat library was used with RepeatMasker. This custom library was created using RepeatModeler [1].

Low complexity features, ab initio predictions and BLAST analyses

Transcription start sites are predicted using Eponine–scan [4]. CpG islands longer than 400 bases and tRNAs are also predicted. The results of Eponine-scan, CpG, and tRNAscan [5] are for display purposes only; they are not used in the gene annotation process.

Section 2: Protein-Coding Model Generation

Various sources of transcript and protein data are investigated and used to generate gene models using a variety of techniques. The data and techniques employed to generate models are outlined here. The numbers of gene models generated are described in gene summary.

Species specific cDNA and protein alignments

cDNAs are downloaded from ENA (www.ebi.ac.uk/ena) and RefSeq [10], and aligned to the genome using Exonerate [11]. Only known mRNAs are used (NMs). The cDNAs can be used to add UTR to the protein coding transcript models if they have a matching set of introns.
Proteins are downloaded from UniProt and filtered based on protein existence (PE) at protein level and transcript level. The proteins are aligned to the genome using PMATCH to reduce the search space, then with genewise, which is a splice-aware aligner, to generate spliced models.

Projection mapping pipeline

For all species we generated a whole genome alignment against a suitable reference assembly using LastZ [12]. Syntenic regions identified using this alignment are then used to map protein-

coding annotation from the most recent closely related species or the Ensembl/GENCODE [13] gene set.
For the *Xenopus tropicalis* annotation, we used the human assembly GRCh38.p12 as a reference to map protein-coding annotation. For each protein-coding gene in human, we projected the coding exons within the canonical transcript to the *Xenopus tropicalis*. In case of exonic overlap on the projected sequence, the longest exon took precedence. If the mapping did not succeed, we selected the next successful projection of the transcript having the longest translation.

Protein-to-genome pipeline

Protein sequences are downloaded from UniProt and aligned to the genome in a splice aware manner using GenBlast [12]. The set of proteins aligned to the genome is a subset of UniProt proteins used to provide a broad, targeted coverage of the reptile proteome. The set consists of the following:

- Self SwissProt/TrEMBL PE 1 & 2
- Self SwissProt/TrEMBL PE 3
- Vertebrates SwissProt/TrEMBL PE 1 & 2

Note: PE level = protein existence level

For the *Xenopus tropicalis* annotation, a cut-off of 50 percent coverage and 30 percent identity and an e-value of e-1 were used for GenBlast with the exon repair option turned on. The top 5 transcript models built by GenBlast for each protein passing the cut-offs are kept.

RNA-seq pipeline

Where available, RNA-seq data is downloaded from ENA (https://www.ebi.ac.uk/ena/) and used in the annotation. A merged file containing reads from all tissues/samples is created. The merged data is less likely to suffer from model fragmentation due to read depth. The available reads are aligned to the genome using BWA [13], with a tolerance of 50 percent mismatch to allow for intron identification via split read alignment. Initial models generated from the BWA alignments are further refined via exonerate. Protein-coding models are identified via a BLAST alignment of the longest ORF against the UniProt vertebrate PE 1 & 2 data set.
In the case where multiple tissues/samples are available we create a gene track for each such tissue/sample that can be viewed in the Ensembl browser and queried via the API.

Immunoglobulin and T-cell Receptor genes

Translations of different human IG gene segments are downloaded from the IMGT database [14] and aligned to the genome using GenBlast.

For the *Xenopus tropicalis* annotation, a cut-off of 80 percent coverage, 70 percent identity and an e-value of e-1 were used for GenBlast with the exon repair option turned on. The top 10 transcript models built by GenBlast for each protein passing the cut-offs are kept.

Selenocysteine proteins

We aligned known selenocysteine proteins against the genome using Exonerate. Then we checked that the generated model had a selenocysteine in the same positions as the known protein. We only kept models with at least 90% coverage and 95% identity.

Section 3: Filtering the Protein-Coding Models

The filtering phase decides the subset of protein-coding transcript models, generated from the model-building pipelines, that comprise the final protein-coding gene set. Models are filtered based on information such as what pipeline was used to generate them, how closely related the data are to the target species and how good the alignment coverage and percent identity to the original data are.

Prioritising models at each locus

The LayerAnnotation module is used to define a hierarchy of input data sets, from most preferred to least preferred. The output of this pipeline includes all transcript models from the highest ranked input set. Models from lower ranked input sets are included only if their exons do not overlap a model from an input set higher in the hierarchy.
Note that models cannot exist in more than one layer. For UniProt proteins, models are also separate into clades, to help selection during the layering process. Each UniProt protein is in one clade only, for example mammal proteins are present in the mammal clade and are not present in the vertebrate clade to avoid aligning the proteins multiple times.
When selecting the model or models kept at each position, we prioritise based on the highest layer with available evidence. In general, the highest layers contain the set of evidence containing the most trustworthy evidence in terms of both alignment/mapping quality, and also in terms of relevance to the species being annotated. So, for example, when a reptile is being annotated, well aligned evidence from either the species itself or other closely related vertebrates would be chosen over evidence from more distant species. Regardless of what species is being annotated, well-aligned human proteins are usually included in the top layer as human is the current most complete vertebrate annotation. For further details on the exact layering used please refer to section 6.

Addition of UTR to coding models

The set of coding models is extended into the untranslated regions (UTRs) using RNA-seq data (if available) and alignments of species-specific RefSeq cDNA sequences. The criteria for adding UTR from cDNA or RNA-seq alignments to protein models lacking UTR (such as the projection models or the protein-to-genome alignment models) is that the intron coordinates from the model missing UTR exactly match a subset of the coordinates from the UTR donor model.

Generating multi-transcript genes

The above steps generate a large set of potential transcript models, many of which overlap one another. Redundant transcript models are collapsed and the remaining unique set of transcript models are clustered into multi-transcript genes where each transcript in a gene has at least one coding exon that overlaps a coding exon from another transcript within the same gene.

Pseudogenes

Pseudogenes are annotated by looking for genes with evidence of frame-shifting or lying in repeat heavy regions. Single exon retrotransposed pseudogenes are identified by searching for a multi-exon equivalent elsewhere in the genome. A total number of genes that are labelled as pseudogenes or processed pseudogenes will be included in the core db, please check Final Gene set Summary.

Section 4: Creating the Final Gene Set

Small ncRNAs

Small structured non-coding genes are added using annotations taken from RFAM [15] and miRBase [16]. NCBI-BLAST was run for these sequences and models built using the Infernal software suite [17].

Cross-referencing

Before public release the transcripts and translations are given external references (cross-references to external databases). Translations are searched for signatures of interest and labelled where appropriate.

Stable Identifiers

Stable identifiers are assigned to each gene, transcript, exon and translation. When annotating a species for the first time, these identifiers are auto-generated. In all subsequent annotations for a species, the stable identifiers are propagated based on comparison of the new gene set to the previous gene set.

Section 5: Final Gene Set Summary

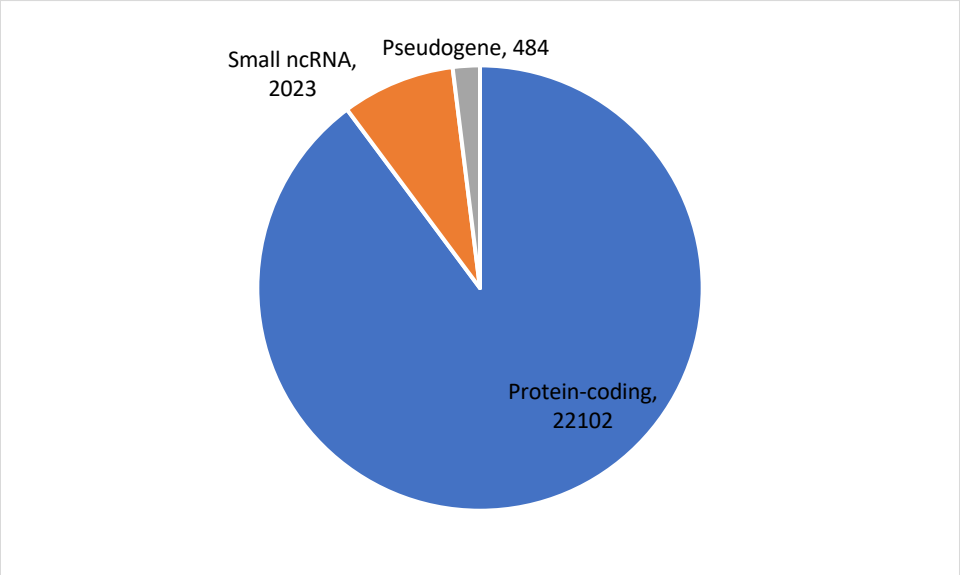

Figure 2: Counts of the major gene classes in each species

Section 6: Appendix - Further information

The Ensembl gene set is generated automatically, meaning that gene models are annotated using the Ensembl gene annotation pipeline. The main focus of this pipeline is to generate a conservative set of protein-coding gene models, although non-coding genes and pseudogenes may also be annotated.
Every gene model produced by the Ensembl gene annotation pipeline is supported by biological sequence evidence (see the "Supporting evidence" link on the left-hand menu of a Gene page or Transcript page); ab initio models are not included in our gene set. Ab initio predictions and the full set of cDNA and EST alignments to the genome are available on our website.
The quality of a gene set is dependent on the quality of the genome assembly. Genome assembly can be assessed in a number of ways, including:

1. Coverage estimates
    - A higher coverage usually indicates a more complete assembly.
    - Using Sanger sequencing only, a coverage of at least 2x is preferred.
2. N50 of contigs and scaffolds
    - A longer N50 usually indicates a more complete genome assembly.
    - Bearing in mind that an average human gene may be 10-15 kb in length, contigs shorter than this length will be unlikely to hold full-length gene models.
3. Number of contigs and scaffolds
    - A lower number top level sequences usually indicates a more complete genome assembly.
4. Alignment of cDNAs and ESTs to the genome
    - A higher number of alignments, using stringent thresholds, usually indicates a more complete genome assembly.

Assembly Information

Species	Common name	Assembly	Genbank accession	Date released
Xenopus tropicalis	Tropical clawed frog	UCB_Xtro_10.0	GCA_000004195.4	2019-11

Table 1: Assembly information

Layers in detail

Layer 1
'IG_C_gene', 'IG_J_gene', 'IG_V_gene', 'IG_D_gene', 'TR_C_gene', 'TR_J_gene', 'TR_V_gene', 'TR_D_gene', 'seleno_self'

Layer 2
'cdna2genome', 'edited', 'gw_gtag', 'gw_nogtag', 'gw_exo', 'rnaseq_merged_1', 'rnaseq_merged_2', 'rnaseq_merged_3', 'rnaseq_merged_4', 'rnaseq_tissue_1', 'rnaseq_tissue_2', 'rnaseq_tissue_3', 'rnaseq_tissue_4', 'self_pe12_sp_1', 'self_pe12_tr_1', 'self_pe12_sp_2', 'self_pe12_tr_2'

Layer 3
'projection_1', 'projection_2', 'rnaseq_merged_5', 'rnaseq_tissue_5', 'reptiles_pe12_sp_1', 'reptiles_pe12_tr_1', 'reptiles_pe12_tr_2', 'reptiles_pe12_sp_2', 'genblast_rnaseq_top'

Layer 4
'human_pe12_sp_1', 'human_pe12_tr_1', 'human_pe12_tr_2', 'human_pe12_sp_2', 'aves_pe12_sp_1', 'aves_pe12_tr_1', 'aves_pe12_tr_2', 'aves_pe12_sp_2', 'amphibians_sp_1', 'amphibians_tr_1', 'amphibians_sp_2', 'amphibians_tr_2', 'reptiles_sp_3', 'reptiles_tr_3', 'reptiles_sp_4', 'reptiles_tr_4', 'genblast_rnaseq_high'

Layer 5
'projection_3', 'projection_4', 'mammals_pe12_sp_1', 'mammals_pe12_tr_1', 'mammals_pe12_sp_2', 'mammals_pe12_tr_2', 'self_pe3_sp_1', 'self_pe3_tr_1', 'genblast_rnaseq_medium'

Layer 6
'human_pe12_sp_3', 'human_pe12_tr_3', 'human_pe12_sp_4', 'human_pe12_tr_4', 'aves_pe12_sp_3', 'aves_pe12_tr_3', 'aves_pe12_sp_4', 'aves_pe12_tr_4', 'amphibians_sp_3', 'amphibians_tr_3', 'amphibians_sp_4', 'amphibians_tr_4'

Layer 7
'mammals_pe12_sp_3', 'mammals_pe12_tr_3', 'mammals_pe12_sp_4', 'mammals_pe12_tr_4', 'human_pe12_sp_5', 'human_pe12_tr_5'

Layer 8
'human_pe12_sp_6', 'human_pe12_tr_6', 'mammals_pe12_sp_5', 'mammals_pe12_tr_5', 'aves_pe12_sp_5', 'aves_pe12_tr_5', 'amphibians_sp_5', 'amphibians_tr_5', 'reptiles_sp_5', 'reptiles_tr_5', 'rnaseq_merged_6', 'rnaseq_tissue_6', 'human_pe12_sp_int_1', 'human_pe12_tr_int_1', 'human_pe12_sp_int_2', 'human_pe12_tr_int_2', 'human_pe12_sp_int_3', 'human_pe12_tr_int_3', 'human_pe12_sp_int_4',

'human_pe12_tr_int_4', 'amphibians_pe12_sp_int_1', 'amphibians_pe12_tr_int_1', 'amphibians_pe12_sp_int_2', 'amphibians_pe12_tr_int_2', 'amphibians_pe12_sp_int_3', 'amphibians_pe12_tr_int_3', 'amphibians_pe12_sp_int_4', 'amphibians_pe12_tr_int_4', 'aves_pe12_sp_int_1', 'aves_pe12_tr_int_1', 'aves_pe12_sp_int_2', 'aves_pe12_tr_int_2', 'aves_pe12_sp_int_3', 'aves_pe12_tr_int_3', 'aves_pe12_sp_int_4', 'aves_pe12_tr_int_4', 'reptiles_pe12_sp_int_1', 'reptiles_pe12_tr_int_1', 'reptiles_pe12_sp_int_2', 'reptiles_pe12_tr_int_2', 'reptiles_pe12_sp_int_3', 'reptiles_pe12_tr_int_3', 'reptiles_pe12_sp_int_4', 'reptiles_pe12_tr_int_4', 'mammals_pe12_sp_int_1', 'mammals_pe12_tr_int_1', 'mammals_pe12_sp_int_2', 'mammals_pe12_tr_int_2', 'mammals_pe12_sp_int_3', 'mammals_pe12_tr_int_3', 'mammals_pe12_sp_int_4', 'mammals_pe12_tr_int_4', 'projection_1_noncanon', 'projection_2_noncanon', 'projection_3_noncanon', 'projection_4_noncanon', 'projection_1_pseudo', 'projection_2_pseudo', 'projection_3_pseudo', 'projection_4_pseudo'

Layer 9
'rnaseq_merged_7', 'rnaseq_tissue_7'

Layer 10
'rnaseq_merged', 'rnaseq_tissue'

More information

More information on the Ensembl automatic gene annotation process can be found at:

- Publication

Aken B et al.: The Ensembl gene annotation system. Database 2016.

- Web

Link to Ensembl gene annotation documentation

References

1. Smit, A., R. Hubley, and P. Green, http://www.repeatmasker.org/. RepeatMasker Open, 1996. **3**: p. 1996-2004.
2. Kuzio, J., R. Tatusov, and D. Lipman, *Dust.* Unpublished but briefly described in: Morgulis A, Gertz EM, Schäffer AA, Agarwala R. A Fast and Symmetric DUST Implementation to Mask Low-Complexity DNA Sequences. Journal of Computational Biology, 2006. **13**(5): p. 1028-1040.
3. Benson, G., *Tandem repeats finder: a program to analyze DNA sequences.* Nucleic acids research, 1999. **27**(2): p. 573.
4. Down, T.A. and T.J. Hubbard, *Computational detection and location of transcription start sites in mammalian genomic DNA.* Genome research, 2002. **12**(3): p. 458-461.
5. Lowe, T.M. and S.R. Eddy, *tRNAscan-SE: a program for improved detection of transfer RNA genes in genomic sequence.* Nucleic acids research, 1997. **25**(5): p. 955-964.
6. Burge, C. and S. Karlin, *Prediction of complete gene structures in human genomic DNA.* Journal of molecular biology, 1997. **268**(1): p. 78-94.
7. Consortium, U., *UniProt: the universal protein knowledgebase.* Nucleic acids research, 2017. **45**(D1): p. D158-D169.
8. Pontius, J.U., L. Wagner, and G.D. Schuler, *21. UniGene: A unified view of the transcriptome.* The NCBI Handbook. Bethesda, MD: National Library of Medicine (US), NCBI, 2003.
9. Altschul, S.F., et al., *Basic local alignment search tool.* Journal of molecular biology, 1990. **215**(3): p. 403-410.
10. O'Leary, N.A., et al., *Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation.* Nucleic acids research, 2015. **44**(D1): p. D733-D745.
11. Slater, G.S.C. and E. Birney, *Automated generation of heuristics for biological sequence comparison.* BMC bioinformatics, 2005. **6**(1): p. 31.

12. She, R., et al., *genBlastG: using BLAST searches to build homologous gene models.* Bioinformatics, 2011. **27**(15): p. 2141-2143.
13. Li, H. and R. Durbin, *Fast and accurate short read alignment with Burrows–Wheeler transform.* Bioinformatics, 2009. **25**(14): p. 1754-1760.
14. Lefranc, M.-P., et al., *IMGT®, the international ImMunoGeneTics information system® 25 years on.* Nucleic acids research, 2014. **43**(D1): p. D413-D422.
15. Griffiths-Jones, S., et al., *Rfam: an RNA family database.* Nucleic acids research, 2003. **31**(1): p. 439-441.
16. Griffiths-Jones, S., et al., *miRBase: microRNA sequences, targets and gene nomenclature.* Nucleic acids research, 2006. **34**(suppl_1): p. D140-D144.
17. Nawrocki, E.P. and S.R. Eddy, *Infernal 1.1: 100-fold faster RNA homology searches.* Bioinformatics, 2013. **29**(22): p. 2933-2935.
18. Finn, R.D., et al., *The Pfam protein families database: towards a more sustainable future.* Nucleic acids research, 2016. **44**(D1): p. D279-D285.