Predicting Chroma from Luma in AV1

Luc N. Trudeau*†, Nathan E. Egge*†, and David Barr†

*Mozilla
331 E Evelyn Ave
Mountain View, CA, 94041, USA
luc@trud.ca, negge@mozilla.com

†Xiph.Org Foundation
21 College Hill Road
Somerville, MA, 02144, USA
b@rr-dav.id.au

Abstract

Chroma from luma (CfL) prediction is a new and promising chroma-only intra predictor that models chroma pixels as a linear function of the coincident reconstructed luma pixels. In this paper, we present the CfL predictor adopted in Alliance Video 1 (AV1), a royalty-free video codec developed by the Alliance for Open Media (AOM). The proposed CfL distinguishes itself from prior art not only by reducing decoder complexity, but also by producing more accurate predictions. On average, CfL reduces the BD-rate, when measured with CIEDE2000, by 4.87% for still images and 2.41% for video sequences.

1 Introduction

Still image and video compression is typically not performed using red, green, and blue (RGB) color primaries, but rather with a color space that separates luma from chroma. There are many reasons for this, notably that luma and chroma are less correlated than RGB, which favors compression; and also that the human visual system is less sensitive to chroma allowing one to reduce the resolution in the chromatic planes, a technique known as chroma subsampling [1].

Another way to improve compression in still images and videos is to subtract the pixels by a predictor. When this predictor is derived from previously reconstructed information inside the current frame, it is referred to as an intra prediction tool. In contrast, an inter prediction tool uses information from previously reconstructed frames. For example, "DC" prediction is an intra prediction tool that predicts the pixels values in a block by averaging the values of neighboring pixels adjacent to the above and left borders of the block [2].

Chroma from luma (CfL) prediction is a new and promising chroma-only intra predictor that models chroma pixels as a linear function of the coincident reconstructed luma pixels [3]. It was proposed for the HEVC video coding standard [4], but was ultimately rejected, as the decoder model fitting caused a considerable complexity increase. It was proposed again as part of the HEVC Range Extension [5], this time without decoder model fitting in order to reduce decoder complexity.

More recently, CfL prediction was implemented in the Thor codec [6] as well as in the Daala codec [7]. The inherent conceptual differences in the Daala codec, when compared to HEVC, led to multiple innovative contributions by Egge and Valin [7] to CfL prediction. Most notably a frequency domain implementation.

As both Thor and Daala served as bases for AV1, a research initiative was established regarding CfL, the results of which are presented in this paper. The proposed

2375-0359/18/$31.00 ©2018 IEEE
DOI 10.1109/DCC.2018.00046

CfL implementation, outlined in Section 3, not only builds on the innovations of [7], but does so in a way that is compatible with the more conventional compression tools found in AV1. This new implementation is considerably different from its predecessors. Its key contributions are:

- Enhanced parameter signaling, as described in Section 6, when compared with [5], the proposed signaling is more precise and finding the RD-optimal parameters is less complex.
- Model fitting the "AC" contribution of the reconstructed luma pixels, as shown in Section 4, which simplifies the model and allows for a more precise fit.
- Chroma "DC" prediction for "DC" contribution, which requires no signaling and, as described in Section 5.

Finally, Section 7 presents detailed results of the compression gains of the proposed CfL prediction implementation in AV1.

2 State of the Art in Chroma from Luma Prediction

As described in [3], CfL prediction models chroma pixels as a linear function of the coincident reconstructed luma pixels. More precisely, let L be an $M \times N$ matrix of pixels in the luma plane; we define C to be the chroma pixels spatially coincident to L. Since L is not available to the decoder, the reconstructed luma pixels, L^{r}, corresponding to L are used instead. The chroma pixel prediction, C^{p}, produced by CfL uses the following linear equation:

$$C^{\mathrm{p}} = \alpha \times L^{\mathrm{r}} + \beta \,. \tag{1}$$

Some implementations of CfL [3, 4, 6] determine the linear model parameters α and β using linear least-squares regression

$$\alpha = \frac{(M \times N) \sum_i \sum_j L^{\mathrm{r}}_{ij} C_{ij} - \sum_i \sum_j L^{\mathrm{r}}_{ij} \sum_i \sum_j C_{ij}}{(M \times N) \sum_i \sum_j \left(L^{\mathrm{r}}_{ij}\right)^2 - \left(\sum_i \sum_j L^{\mathrm{r}}_{ij}\right)^2} \,, \tag{2}$$

$$\beta = \frac{\sum_i \sum_j C_{ij} - \alpha \sum_i \sum_j L^{\mathrm{r}}_{ij}}{M \times N} \,. \tag{3}$$

We classify [3, 4, 6] as implicit implementations of CfL, since α and β are not signaled in the bitstream, but are implied from the bitstream. The main advantage of the implicit implementation is the absence of signaling.

However, implicit implementations have numerous disadvantages. As mentioned before, computing least squares considerably increases decoder complexity. Another important disadvantage is that the chroma pixels, C, are not available when computing least squares on the decoder. As such, prediction error increases since neighboring reconstructed chroma pixels must be used instead.

In [7], the authors argue that the advantages of explicit signaling considerably outweigh the signaling cost. This is corroborated by the results in [5]. Based on these findings, we propose a hybrid approach that signals α and implies β.

3 The Proposed Chroma from Luma Prediction

This section outlines the proposed chroma-only intra predictor. To predict chroma samples, CfL starts with the spatially coinciding reconstructed luma pixels.

As illustrated in Fig. 1, when chroma subsampling is used, in order for the pixels to coincide, the reconstructed luma pixels are subsampled accordingly. As explained in Section 4, the coinciding reconstructed luma pixels are subtracted by their average, which results in their "AC" contribution.

As for the scaling factor indices and signs, they are decoded from the bitstream, which is described in Section 6. The CfL prediction is built by multiplying the "AC" contribution of reconstructed luma pixels with the scaling factors and the result is added to the intra "DC" prediction, as explained in Section 5.

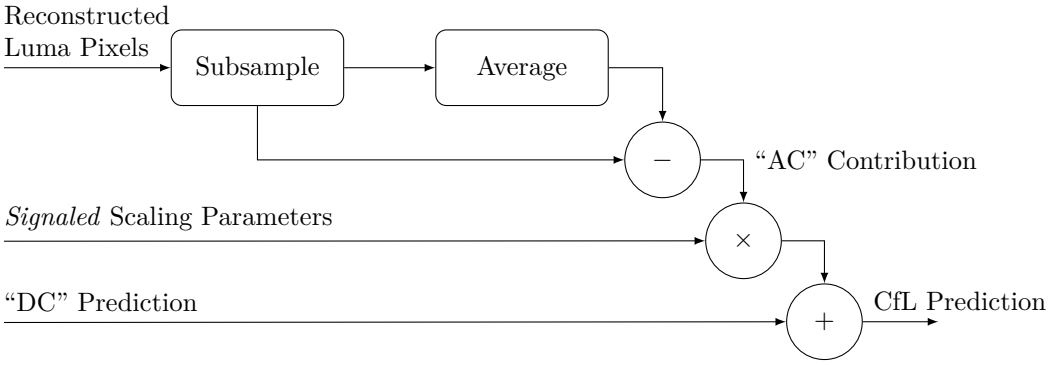

Figure 1: Outline of the operations required to build the proposed CfL prediction.

4 Model Fitting the "AC" Contribution

In [7], Egge and Valin demonstrate the merits of separating the "DC" and "AC" contributions of the frequency domain CfL prediction. In the pixel domain, the "AC" contribution of a block can be obtained by subtracting it by its average.

An important advantage of the "AC" contribution is that it is zero mean, which results in significant simplifications to the least squares model parameter equations. More precisely, let L^{AC} be the zero-meaned reconstructed luma pixels. Because $\sum_i \sum_j L^{\mathrm{AC}} = 0$, substituting L^{r} by L^{AC} yields the following simplified model parameters equations:

$$\alpha^{\mathrm{AC}} = \frac{\sum_i \sum_j L_{ij}^{\mathrm{AC}} C_{ij}}{\sum_i \sum_j \left(L_{ij}^{\mathrm{AC}}\right)^2}\,, \tag{4}$$

$$\beta^{\mathrm{AC}} = \frac{\sum_i \sum_j C_{ij}}{M \times N}\,. \tag{5}$$

We define the zero-mean chroma prediction, C^{AC}, like so

$$C^{\mathrm{AC}} = \alpha^{\mathrm{AC}} \times L^{\mathrm{AC}} + \beta^{\mathrm{AC}}\,. \tag{6}$$

When computing the zero-mean reconstructed pixels, the resulting values are stored using 1/8th precision fixed-point values. This ensures that even with 12-bit integer pixels, the average can be stored in a 16-bit signed integer.

Traditionally, subsampling is performed by adding the coincident pixels in the luma plane and dividing by the number of pixels. The exact number of coincident pixels is determined by the type subsampling. AV1 supports: 4:2:0, 4:2:2, 4:4:0 and 4:4:4 chroma subsamplings [1]. Let $s_x, s_y \in \{1, 2\}$ be the subsampling steps along the x and y axes, respectively. It follows that the summing the coincident pixels at position (u, v) is performed as follows:

$$\mathrm{S}(s_x, s_y, u, v) = \sum_{y=1}^{s_y} \sum_{x=1}^{s_x} L^{\mathrm{r}}_{s_y \times v + y, s_x \times u + x} \tag{7}$$

This luma subsampling step was considered too costly for HEVC [4] which explains why [5] is only available for 4:4:4. We propose a simpler subsampling scheme that is less complex and more precise.

By combining the luma subsampling step with the average subtraction step (shown in Fig. 1), not only do the equations simplify, but the subsampling divisions and the corresponding rounding error are removed. The equation corresponding to the combination of both steps is given in Eq. (8), which simplifies to Eq. (9). Note that both equations use integer divisions.

$$L^{\mathrm{AC}}_{u,v} = 8 \left(\frac{\mathrm{S}(s_x, s_y, u, v)}{s_y \times s_x} \right) - \frac{8 \sum_i \sum_j \left(\frac{\mathrm{S}(s_x, s_y, i, j)}{s_y \times s_x} \right)}{M \times N} \tag{8}$$

$$\implies \frac{1}{s_y \times s_x} \left(8 \times \mathrm{S}(s_x, s_y, u, v) - \frac{\sum_i \sum_j 8 \times \mathrm{S}(s_x, s_y, i, j)}{M \times N} \right) \tag{9}$$

Based on the supported chroma subsamplings, it can be shown that $s_y \times s_x \in \{1, 2, 4\}$ and that since both M and N are powers of two, $M \times N$ is also a power of two. It follows that both $\frac{1}{s_y \times s_x}$ and $\frac{1}{M \times N}$ in Eq. (9) can be replaced by bit shift operations.

For example, in the context of a 4:2:0 chroma subsampling, instead of applying a box filter, the proposed approach only requires to sum the 4 reconstructed luma pixels that coincide with the chroma pixels. Afterwards, when CfL will scale its luma pixels to improve the precision of the predictions, [5] requires to scale by 8, whereas the proposed approach only needs to scale by 2 (i.e. $\frac{8}{2 \times 2}$). Both approach are now on the same scale but the rounding errors saved in Eq. (9) results in more precise values for the proposed approach. In other words, we will perform the integer division required by the box filter only when we downscale the predicted pixel values.

5 Chroma "DC" Prediction for "DC" Contribution

Switching the linear model to use zero mean reconstructed luma pixels also changes the "DC" contribution, to the extent that it now only depends on C. This can be seen in Eq. (5), where β^{AC} is the average of the chroma pixels.

The chroma pixel average for a given block is not available in the decoder. However, there already exists an intra prediction tool that predicts this average. When applied to the chroma plane, the "DC" prediction predicts the pixel values in a block by averaging the values of neighboring pixels adjacent to the above and left borders of the block [2]. In Fig. 2, we present an analysis of the "DC" prediction error over the Kodak True Color Image suite.

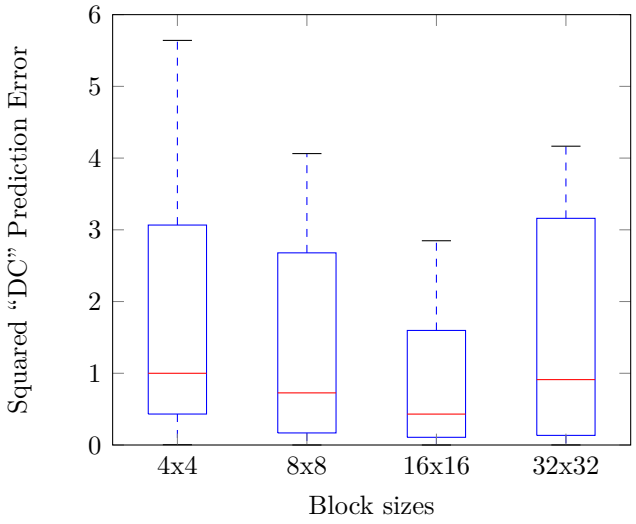

Figure 2: Error analysis of the "DC" predictor.

Note that Fig. 2 does not include outliers, as they hinder readability of the figure. Furthermore, it is unlikely that the intra mode selection algorithm would select CfL to predict such content. The median squared "DC" prediction error is equal to or less than 1 for all tested block sizes.

Based on this error analysis, the absence of signaling and the fact that "DC" prediction is already implemented in AV1, we use "DC" prediction as an approximation for β^{AC}, as shown in Fig. 1. The proposed CfL prediction is expressed as follows:

$$\mathrm{CfL}(\alpha) = \alpha \times L^{\mathrm{AC}} + \mathrm{DC}\,. \tag{10}$$

6 Parameter Signaling

Signaling the scaling parameters allows encoder-only fitting of the linear model. This reduces decoder complexity and results in a more precise prediction, as the best scaling parameter can be determined based on the reference chroma pixels which are only available to the encoder.

Signaling the scaling parameters fundamentally changes their selection. In this context, the least-squares regression used in [3, 4, 6] does not yield an RD-optimal solution as it ignores the trade-off between the rate and the distortion of the scaling

parameters. For the proposed CfL prediction, the signaling parameters are determined using the same rate-distortion optimization mechanics as other coding tools and parameters of AV1. Concretely, given a set of scaling parameters A, the selected scaling parameter is the one that minimizes the trade-off between the rate and the distortion

$$\alpha = \arg\min_{a \in A} \left(D(\mathrm{CfL}(a)) + \lambda R(a) \right) \ . \tag{11}$$

In the previous equation, the distortion, D, is the sum of the squared error between the reconstructed chroma pixels and the reference chroma pixels. Whereas, the rate, R, is the number of bits required to encode the scaling parameter and the residual coefficients. Furthermore, λ is the weighing coefficient between rate and distortion used by AV1.

CfL parameters are signaled at the prediction unit level, with consideration to the fact that rate-distortion optimization approaches are used over the traditional least squares regression. Since CfL is an chroma-only intra prediction mode, there is no need to always signal a skip flag, when CfL is not desired as is the case for [5].

CfL parameters apply to the whole prediction unit. The "DC" prediction, used by CfL, is computed over the entire prediction unit. This greatly reduces the rate-distortion search space of CfL parameters as they do not interact with the interdependencies of the transform blocks inside a prediction unit.

In [5], CfL parameter signaling is at the transform block level. This creates more interdependencies between transform blocks as the CfL parameters will change the pixels used when performing intra prediction on neighboring transform blocks. Evaluating all these combinations is prohibitively expensive resulting in the use of fast approximation approaches that cannot guarantee optimal results. The proposed solution avoid these issues, with the added benefit that reducing interdependencies between transform blocks speeds up the prediction and reconstruction process of prediction units.

When the CfL chroma only mode is chosen, we first signal the joint sign of both scaling parameters. A sign is either negative, zero, or positive. Contrary to [5], the proposed signaling does not permit choosing (zero, zero), as it results in "DC" prediction. It follows that the joint sign requires an eight-value symbol.

As for each scaling parameter, a 16-value symbol is used to represent values ranging from 0 to 2 with a step of 1/8th. The entropy coding details are beyond the scope of this paper; however, it is important to note that a 16-value symbol fully utilizes the capabilities of the multi-symbol entropy encoder [8]. In comparison with [5], the proposed signaling scheme offers twice the range and twice the precision. Finally, scaling parameters are signaled only if they are non-zero.

7 Experimental Results

To ensure a valid evaluation of coding efficiency gains, our testing methodology conforms to that of [9]. All simulation parameters and a detailed sequence-by-sequence breakdown for all the results presented in this paper are available online at [10]. Fur-

Table 1: Results over the Subset1 test set (still images), available online [18].

	BD-Rate						
	PSNR	PSNR-HVS	SSIM	**CIEDE2000**	PSNR Cb	PSNR Cr	MS SSIM
Average	-0.53	-0.31	-0.34	**-4.87**	-12.87	-10.75	-0.34

Table 2: Results over the Objective-1-fast test set (video sequences), available online [19].

	BD-Rate						
	PSNR	PSNR-HVS	SSIM	**CIEDE2000**	PSNR Cb	PSNR Cr	MS SSIM
Average	-0.43	-0.42	-0.38	**-2.41**	-5.85	-5.51	-0.40
1080p	-0.32	-0.37	-0.28	**-2.52**	-6.80	-5.31	-0.31
1080p-screen	-1.82	-1.72	-1.71	**-8.22**	-17.76	-12.00	-1.75
360p	-0.15	-0.05	-0.10	**-0.80**	-2.17	-6.45	-0.04
720p	-0.12	-0.11	-0.07	**-0.52**	-1.08	-1.23	-0.12

thermore, the bitstreams generated in these simulations can be retrieved and analyzed online at [11].

The following tables show the average percent rate difference measured using the Bjøntegaard rate difference, also known as BD-rate [12]. The BD-rate is measured using the following objective metrics: PSNR, PSNR-HVS [13], SSIM [14], CIEDE2000 [15] and MSSIM [16]. Of all the previous metrics, only the CIEDE2000 considers both luma and chroma planes. It is also important to note that the distance measured by this metric is perceptually uniform [15].

As required in [9], for individual feature changes in libaom, we use quantizers: 20, 32, 43, and 55. We present results for three test sets: Objective-1-fast [9], Subset1 [17] and Twitch [17].

In Table 1, we present the results for the Subset1 test set. This test set contains still images, which are ideal to evaluate the chroma intra prediction gains of CfL when compared to other intra prediction tools in AV1.

For still images, when compared to all of the other intra prediction tools of AV1 combined, CfL prediction reduces the rate by an average of 4.87% for the same level of visual quality measured by CIEDE2000.

For video sequences, Table 2 breaks down the results obtained over the objective-1-fast test set.

Not only does CfL yield better intra frames, which produces a better reference for inter prediction tools, but it also improves chroma intra prediction in inter frames. We observed CfL predictions in inter frames when the predicted content was not available in the reference frames. As such, CfL prediction reduces the rate of video sequences by an average of 2.41% for the same level of visual quality when measured with CIEDE2000.

The average rate reductions for 1080p-screen are considerably higher than those of other types of content. This indicates that CfL prediction considerably outperforms

Table 3: Results over the Twitch test set (gaming screen content), available online [20].

	BD-Rate						
	PSNR	PSNR-HVS	SSIM	**CIEDE2000**	PSNR Cb	PSNR Cr	MS SSIM
Average	-1.01	-0.93	-0.90	**-5.74**	-15.58	-9.96	-0.81

other AV1 predictors for screen content coding. As shown in table 3, the results on the Twitch test set, which contains only gaming-based screen content, corroborates this finding.

The sequence-by-sequence results presented in [20] indicate that CfL prediction is particularly efficient for sequences of the game Minecraft, where the average rate reduction exceeds 20% for the same level of visual quality measured by CIEDE2000.

8 Conclusion

In this paper, we presented the chroma from luma prediction tool adopted in AV1. This new implementation is considerably different from its predecessors. Its key contributions are: parameter signaling, model fitting the "AC" contribution of the reconstructed luma pixels, and chroma "DC" prediction for "DC" contribution. Not only do these contributions reduce decoder complexity, but they also reduce prediction error; resulting in a 4.87% average reduction in BD-rate, when measured with CIEDE2000, for still images, and 2.41% for video sequences. Possible improvements to the proposed solution includes non-linear prediction models and motion-compensated CfL.

Reference to Prior Literature

[1] Y. Wang, Y.-Q. Zhang, and J. Ostermann, *Video Processing and Communications*, 1st ed. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2001.

[2] L. Ze-Nian, M. S. Drew, and J. Liu, *Fundamentals of Multimedia*, 2nd ed. Springer Publishing Company, Incorporated, 2014.

[3] J. Kim, S. Park, Y. Choi, Y. Jeon, and B. Jeon, "New intra chroma prediction using inter-channel correlation," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Tech. Rep. JCTVC-B021, Jul. 2010.

[4] J. Chen, V. Seregin, W.-J. Han, J. Kim, and B. Jeon, "Ce6.a.4: Chroma intra prediction by reconstructed luma samples," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Tech. Rep. JCTVC-E266, Mar. 2011.

[5] W. Pu, W.-S. Kim, J. Chen, K. Rapaka, L. Guo, J. Sole, and M. Karczewicz, "Non-rce1: Inter color component residual prediction," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Tech. Rep. JCTVC-N0266, Jul. 2013.

[6] S. Midtskogen, "Improved chroma prediction," IETF NETVC Internet-Draft, Tech. Rep. draft-midtskogen-netvc-chromapred-02, Oct. 2016.

[7] N. E. Egge and J.-M. Valin, "Predicting chroma from luma with frequency domain intra prediction," *Proceedings of SPIE 9410, Visual Information Processing and Communication VI*, vol. 9410, Mar. 2015.

[8] J.-M. Valin, T. B. Terriberry, N. E. Egge, T. Daede, Y. Cho, C. Montgomery, and M. Bebenita, "Daala: Building a next-generation video codec from unconventional technology," *Multimedia signal processing (MMSP) workshop*, no. arXiv:1608.01947, Sep. 2016.

[9] T. Daede, A. Norkin, and I. Brailovsky, "Video codec testing and quality measurement," IETF NETVC Internet-Draft, Tech. Rep. draft-ietf-netvc-testing-05, Mar. 2017.

[10] Xiph.Org Foundation, "Are We Compressed Yet?" [Online]. Available: https://arewecompressedyet.com

[11] M. Bebenita, "AV1 bitstream analyzer," Mozilla. [Online]. Available: https://arewecompressedyet.com/analyzer/

[12] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," Video Coding Experts Group (VCEG) of ITU-T, Tech. Rep. VCEG-M33, 13th Meeting: Austin, Texas, USA, 2001.

[13] K. Egiazarian, J. Astola, N. Ponomarenko, V. Lukin, F. Battisti, and M. Carli, "Two new full-reference quality metrics based on HVS," in *Proceedings of the Second International Workshop on Video Processing and Quality Metrics for Consumer Electronics, VPQM*, Jan. 2006.

[14] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, Apr. 2004. [Online]. Available: http://dx.doi.org/10.1109/TIP.2003.819861

[15] Y. Yang, J. Ming, and N. Yu, "Color image quality assessment based on ciede2000," *advances in multimedia*, vol. 2012, no. Article ID 273723, 2012, http://dx.doi.org/10.1155/2012/273723.

[16] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *The 37th Asilomar Conference on Signals, Systems Computers*, vol. 2, Nov. 2003, pp. 1398–1402.

[17] T. Daede, "Test sets," hosted by the Xiph.org Foundation. [Online]. Available: https://people.xiph.org/~tdaede/sets/

[18] L. Trudeau, "Results of chroma from luma over the subset1 test set," Are We Compressed Yet?, Nov. 2017. [Online]. Available: https://doi.org/10.6084/m9.figshare.5577661.v2

[19] ——, "Results of chroma from luma over the objective-1-fast test set," Are We Compressed Yet?, Nov. 2017. [Online]. Available: https://doi.org/10.6084/m9.figshare.5577778.v1

[20] ——, "Results of chroma from luma over the twitch test set," Are We Compressed Yet?, Nov. 2017. [Online]. Available: https://doi.org/10.6084/m9.figshare.5577946.v1