Ensembling Unets for rare chromosal aberration detection in metaphase images, uncertainty quantification, and ionizing radiation dose estimation

Antonin Deschemps, Eric Grégoire, Juan S Martinez, Aurélie Vaurijoux, Pascale Fernandez, Delphine Dugue, L Bobyk, Marco Valente, Gaëtan Gruel, Emmanuel Moebel, et al.

Ensembling Unets for rare chromosal aberration detection in metaphase images, uncertainty quantification, and ionizing radiation dose estimation

A. Deschemps, E. Grégoire, J.S. Martinez, A. Vaurijoux, P. Fernandez, D. Dugue, L. Bobyk, M. Valente, G. Gruel, E. Moebel, M.A. Benadjaoud, and C. Kervrann*

Authors

A. Deschemps, E. Moebel, and C. Kervrann are with SAIRPICO Project-Team, Inria Center at University of Rennes, SAIRPICO Team, Cellular and Chemical Biology Unit, U1143 INSERM, UMR3666 CNRS, Institut Curie, PSL Research University, Campus universitaire de Beaulieu, 35042 Rennes, France (email: charles.kervrann@inria.fr, ORCID number: 0000-0001-6263-0452, $*$ corresponding author)

E. Grégoire, J.S. Martinez, A. Vaurijoux, P. Fernandez, D. Dugue, G. Gruel, and M.A. Benadjaoud are with IRSN/PSE-SANTE/SERAMED/LRAcc, Radiobiology and Regenerative Medicine Research Service, Radiobiology of Accidental Exposure Laboratory, Fontenay-aux-Roses, France (email: mohamedamine.benadjaoud@irsn.fr)

L. Bobyk and M. Valente are with the Department of Radiation Biology, Armed Forces Biomedical Research Institute, Brétigny-sur-Orge, France

Abstract

In biological dosimetry a radiation dose is estimated using the average number of chromosomal aberrations per peripheral blood lymphocytes. This analysis is still manually performed on 2D metaphase images depicting the 23 pairs of chromosomes because the false discovery rate of current automated detection systems is too high and variable because of sensitivity to small variations in image quality (chromosome spread, illumination variations ...). Therefore, the current systems are only used to assist human experts. Designing more performant automatic and reliable chromosomal aberration detection systems has become of paramount importance to improve diagnosis speed and reduce human expertise time. Here, we propose a novel deep-learning method for automatic rare chromosomal aberration detection and uncertainty quantification. We formulate the problem as a unique regression problem requiring the minimization of a sparsity-promoting loss to reduce the false alarm rate. Furthermore, we select checkpoints at the end of each epoch during training to form a model ensemble. The resulting artificial experts are further analyzed to derive a consensus voting, similar to an agreement of human annotator rating, to provide trustworthy aberration detections and confidence intervals. A radiation dose curve is finally derived from deep learning-assisted counting of dicentrics and fragments in metaphase images, in high agreement with the reference hand-crafted curve in biological dosimetry.

Keywords

biological dosimetry, chromosome aberration, microscopy image analysis, convolutional neural networks, model aggregation, sparse detection.

1 Introduction

Biological dosimetry aims at estimating ionizing radiation doses from biomarkers. The current gold standard (defined by the IAEA [1]) relies on estimating how frequently dicentric chromosomes (i.e., chromosomes with two centromeres) appear in peripheral blood lymphocytes after an exposure event. Nevertheless, variations in microscopy image acquisition conditions as well as chromosome morphology makes object counting by image analysis methods a challenging problem. Furthermore, the need for an accurate estimation of the average number of dicentric per cell means that a large number of image has to be processed. In real-world scenarios, counting by visual inspection by experts is intrinsically limited; the cognitive load is high and the number of specialists insufficient. Automatic and reliable systems are absolutely necessary to face large-scale exposition challenges.

To observe chromosomal aberrations in microscopy images, blood cells are grown for 48 hours, and Demecolcine is used to stop cell division in the metaphase stage of mitosis [2]. As chromosome are translucent objects, Giemsa staining (ethylene blue, azure and eosin) is used to increase image contrast as the sample is spread on a glass slide for imaging. This is one of the most popular microscopic stains, commonly used in hematology, histology, cytology and bacteriology for *in vitro* diagnostic. At the metaphase stage, the 23 pairs of chromosomes in a single cell and potential aberrations are localized and identified in 2D images acquired with a conventional light microscope (see Fig. 1). Aberration counting is manually performed but still remains a tedious task in Giemsa images [3], even if visual inspection of images is assisted by semi-automatic methods. At the end, a calibration curve is used to convert the aberration yield into an ionizing radiation dose [4]. The aberration yield for an individual who was not exposed to any radiation is expected to be around 1 aberration every 1000 cells. Accordingly, controlling the false discovery rate of aberrations becomes essential to avoid over-diagnosing acute radiation exposition and overloading care facilities, especially in the case of a large-scale exposition.

Since the seminal ImageNet paper [5], convolutional neural networks (ConvNets) based on architectures composed of successive multiscale neuron layers, have brought significant advances in biomedical imaging [6] and microscopy [7;8;9]. In karyotyping analysis, artificial neural networks and deep learning were investigated for chromosome detection and enumeration [10], segmentation [11;12], straightening [13;14], identification [15;16;17;18;19], and aberration simulation [20]. Unlike the aforementioned works focused on chromosome detection and segmentation, the problem of chromosomal aberration detection is not frequently addressed in the literature; only a few of papers tackle the detection of dicentric chromosomes with deep learning methods. The problem of aberration counting in metaphase images is not easy to formulate within the deep learning framework as it must predict zero event in most images and one or two events at most in very few images. So far, deep learning models suffer from over-confidence and mis-calibration [21]; if a classification model predicts a class with a probability of 99%, there is no guarantee that the error rate is close to 1%. The prediction of rare events theoretically requires the building of a very large datasets of "normal" images with no aberration and a few images depicting very few anomalies to be defined. While the recent published deep learning methods, based for instance on object detection convolutional neural networks models [22], are promising methods, none of them addresses the problem of model uncertainty quantification [4;23;24]. Now, uncertainty quantification becomes of paramount importance in biological dosimetry to to build trustworthy deep-learning based models. Moreover, providing trustworthy solutions with normalized uncertainty quantification values is needed to compare the performance of methods, especially when no common benchmark and publicly image database is available. Beyond biological dosimetry, these concerns about uncertainty quantification and statistical analysis deep learning models have recently become central in biomedical imaging analyses [25].

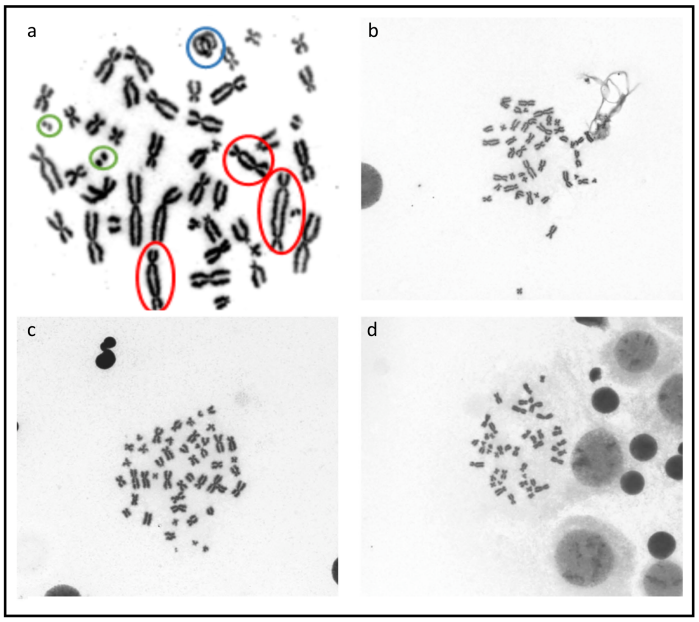


Fig. 1: Giemsa images depicting metaphases. a: Manual labeling of dicentric chromosomes (red circle) and fragments (green circle) ; b-d: Three typical metaphases depicting variable contents (chromosomes, debris, nucleus) acquired with a bright field microscope with different setting inducing variable illumination conditions.

To overcome these drawbacks, we propose here an alternative deep learning approach based on sparse regression and ensembling models that enables to boost the performance of current aberration counting methods and provides confidence intervals. Instead of detecting the 23 chromosomes in metaphase images by applying bounding box regression-like algorithms in the first step and classifying monocentric and dicentric chromosomes in the second step, we formulate the chromosomal aberration detection as a semantic segmentation task. The main advantage of using or a single U-Net network is that it further simplifies uncertainty estimation. As we aim at detecting small rare objects, that is dicentric center-points, the training loss is specifically designed to produce sparse detection maps as very few pixels are expected to be aberration pixels. To quantify uncertainty, we have explored the randomness of model training and build a diverse ensemble. This is achieved here by collecting the models estimated at the end of each epoch during training after a set number of "warmup" epochs. The diversity and dynamics of model ensemble are tracked over epochs and originally

displayed through a low-dimensional projection to highlight the relationships between training stochasticity and ensemble diversity. The U-net models are then aggregated to reach a high level of performance and quantify prediction uncertainty. The method is used to enumerate independently dicentrics and fragments in metaphase images and finally the results are combined to infer a radiation dose calibration curve that reliably fits the manual reference curve used in biological dosimetry. In the experiments, we evaluated the performance and the behavior of the single model and model ensemble, in particular we demonstrate the robustness to distribution shift between the imbalanced training dataset.

2 Results

2.1 Sparse regression improves Precision and Recall of rare aberration detectors

Regression models for model detection amounts to estimating a 2D probability map (or "heatmap") in which the local maxima may be approximated by multiple Gaussian spots corresponding to objects to be detected in the image. Our supervised Unet model[26] is trained to predict Gaussian spot positions in series of downscaled image domains, (and upsampled in the decoder) by a factor of 2 at each layer. The parameters θ of our Unet model are learned by minimizing a loss function composed of a reconstruction term and a sparse-promoting regularization term (see (1)-(4) in Methods). The specific regularizer encourages the emergence of a small number of "hot" spots (i.e., dicentric chromosomes or fragments), assumed to be unusual events in Giemsa images.

Most images routinely analyzed by biologists do not contain any aberration, even for high doses. Our annotated training dataset is a subset of a much larger archive of patient data (see description in Methods). This annotated dataset contains 5,430 images over $\sim$ 80,000 images depicting at least one aberration. This reduces training time considerably, and prevents the discovery of trivial models where no object is ever predicted. Even if our training dataset is not an accurate representation of real-world metaphase images as metaphases containing aberrations are considerably over-represented, the model performance in terms of Precision and Recall scores, defined from True Positives (TP), False Positives (FP), and False Negatives (FN) (see Fig. 2), is higher than the well-established DCScore method in biological dosimetry, that relies on conventional computer vision techniques for segmenting and classifying chromosomes in metaphase images. In[27], it is reported that the Recall of DCScore lies between 35% and 75% after the segmentation step, while the Recall is around 35% and the Precision around 40% after the classification step (see Table 1). DCSore is prone to segmentation errors that generally induce classification errors, especially if a chromosome is split into several components. Overall, the performance of DCScore based on a segmentation-classification procedure is not very robust to illumination variations due to low staining or non-optimal image acquisition quality. Because all chromosomes may not be correctly retrieved (Recall is less than 1), this model tends to underestimate ionizing radiation doses.

As our Unet model is trained to predict a Gaussian spot, the thresholded predictions are binary images comprised of approximately circular spots. Accordingly, the Gaussian spots in the ground truth heatmaps are also converted to binary circular disks. The True Positives are usually defined up to a small location error, as matching the ground truth perfectly would be too stringent. In our case, the spots are small compared to the object size so that the position error remains very small even in the cases where the intersection between the predicted and ground truth spot is the smallest possible one (one pixel, see Fig. 2). Therefore, we consider any overlap between a prediction and a ground truth spot to be a True Positive. Predicted objects that do not overlap ground truth spots are considered to be False Positives. Finally, objects in the ground truth heatmaps that are not predicted by the model are considered to be False Negatives. True Negatives are ill-defined in object detection, and are not considered.

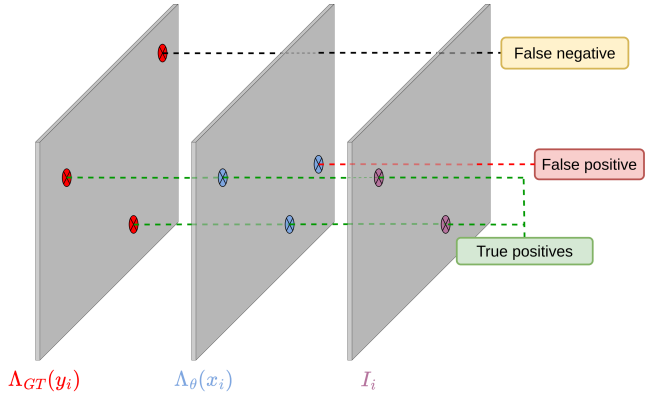


Fig. 2: Sketch of model evaluation. The intersection image $\boldsymbol{I}_i$ between the (binary) ground truth $\boldsymbol{\Lambda}_{GT}(y_i)$ and the (binary) prediction map $\boldsymbol{\Lambda}_\theta(x_i)$ is computed. The number of True Positives, False Positives and False Negatives are defined as follows: $TP = N_{cc}(\boldsymbol{I}_i)$, $FP = \max(0, N_{cc}(\boldsymbol{\Lambda}_\theta(x_i)) - TP)$, and $FN = \max(0, N_{cc}(\boldsymbol{\Lambda}_{GT}(y_i)) - TP)$, where $N_{cc}(\boldsymbol{I}_i)$, $N_{cc}(\boldsymbol{\Lambda}_\theta(x_i))$ and $N_{cc}(\boldsymbol{\Lambda}_{GT}(y_i))$ denote the number of connected components in $\boldsymbol{I}_i$, $\boldsymbol{\Lambda}_\theta(x_i)$ and $\boldsymbol{\Lambda}_{GT}(y_i)$, respectively. In the toy example, we have two True Positives, one False Negative and one False Positive, so that Precision is $TP/(TP+FP) = 2/3$ and Recall is $TP/(TP+FN) = 2/3$

In Tables 1 and 2 and Fig. 3a-b, we reported the Precisions and Recall scores obtained with the L_2 loss (Model A) and the Sparse Variation (SV) loss (Model B) (see 3 in Methods. Models A and B outperformed the results[27] obtained with DCScore in the case of monocentric chromosome versus dicentric chromosome classification. Training with a loss that promotes sparsity significantly improves Precision and Recall for fragments and dicentric chromosomes. Nevertheless, some performance variation is noticeable during training, as confirmed by the inter-quantile range of performance shown in Fig. 3a-b. This variation suggests that there is sufficient parameter space exploration to get enough prediction diversity for aggregation to be worthwhile (see next section). Note that, in biological dosimetry, a high Precision score is desirable to reduce false positives detection rate.

2.2 Ensembling diverse models boost performance and provide confidence intervals

To reduce the number of false positives (i.e., improve Precision) at a fixed Recall level, we investigated the ensemble method[28]. Ensemble learning is commonly used in artificial neural networks

	Precision	Recall
Model A (L_2 loss)	76.6% [68.8 - 84.7]	45.2% [31.6 - 56.4)]
Model B (SV loss)	83.6% [75.4 - 92.2]	51.2% [37.0, 62.7]
Ensemble	**85.8%** [83.7 - 88.3]	**61.7%** [57.1 - 65.8]
DCScore	$\sim 40\%$	$\sim 35\%$

Table 1: Comparison between Model A (L_2 loss, $T_C = 0.6, \lambda = 0$), Model B (Sparse Variation (SV) loss), $T_C = 0.6, \lambda = 0.2, \rho = 0.1$), Ensemble ($T_A = 4$, $T_C = 0.5$), and DCScore for the dicentric class. The inter-quantile intervals $[q_{5\%} - q_{95\%}]$ are reported in square brackets. All performance scores are computed on a separate test set. DCScore performance was reported in [40].

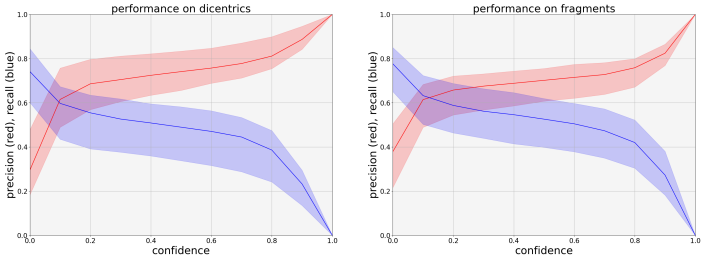
	Precision	Recall
Model A (L_2 loss)	70.8% [62.0 - 77.4]	49.4% [37.8 - 59.7]
Model B (SV loss)	79.3% [71.8 - 84.2]	55.0 % [46.0 - 66.3]
Ensemble	**81.0%** [78.5 - 83.3]	**65.5 %** [62.4 - 68.5]

Table 2: Comparison between Model A (L_2 loss, $T_C = 0.6, \lambda = 0$), Model B (Sparse Variation (SV) loss, $T_C = 0.6, \lambda = 0.2, \rho = 0.1$) and Ensemble ($T_A = 4, T_C = 0.5$) for the fragment class. The inter-quantile intervals $[q_{5\%} - q_{95\%}]$ are reported in square brackets. For Model A and Model B, the confidence interval is computed over the last 50 epochs of training. For the Ensemble, it is computed over a 100 randomly sampled ensembles (i.e., checkpoints). All performance scores are computed on the test set.

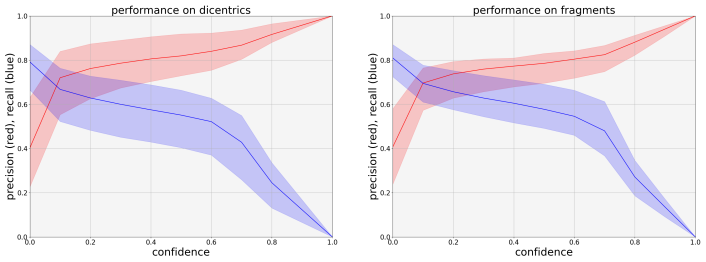
for uncertainty modeling [25;28] and performance improvements [29;30;31]. For instance, ensembling semantic segmentation models to provide spatial uncertainty measurements based on differences in model prediction is well established in the literature (e.g., see Devan et al. [32]). We have adapted a particular strategy that consists here in emulating the decision process of several artificial experts as explained below.

Several informative and non redundant models can be strategically selected during training until convergence. The checkpoints are typically set at the end of each epoch to avoid excessive autocorrelation between samples, and to form a collection of models with a high diversity but with the same dimensionality. Because training a deep neural network is a stochastic process, it is well established that successive training runs of the same model tend to explore different regions of the parameter space. Those local minima are usually very close in terms of validation loss, but their predictions are not identical [33;34]. As shown in [33], it is not necessary to run several successive training runs. A carefully chosen learning rate schedule may be enough to achieve efficient parameter space exploration, and further to build a diverse ensemble from checkpoints of a single training run.

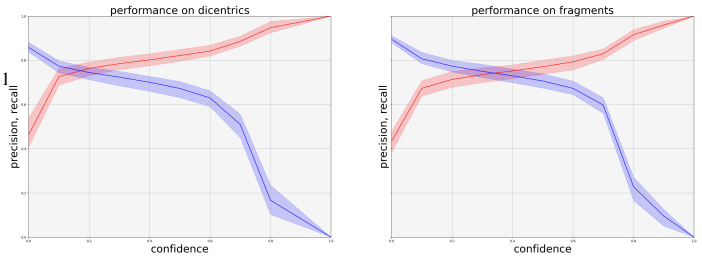
This diversity of training trajectories and learning dynamics in feature space is here originally displayed on a 2D maps via a novel low-dimensional principal component analysis (PCA)-based representation. This visualization technique ((see Methods) is very helpful and informative as it shows that models collected at the end of each epoch capture informative and variable features. Classification networks output a single classification vector per image, so that plotting training dynamics over time using dimensionality reduction is relatively easy [35]. A scatterplot of classification vectors embedded in a lower dimension at each epoch provides a good view of how classes are progressively separated during training. Unlike the previous approach [35], we consider feature maps as bags of independent (one for each vector) feature vectors as illustrated in Fig. 4. We do not consider feature



(a) Performance of Model A (L_2 loss) (dicentric chromosomes (left), fragments (right))



(b) Performance of Model B (Sparse Variation loss) (dicentric chromosomes (left), fragments (right))



(c) Performance of the ensemble of Model B (Sparse Variation loss) (dicentric chromosomes (left), fragments (right))

Fig. 3: **Precision and Recall as functions of confidence for dicentric chromosomes (left column) and fragments (right column)**. Top: performance summary of Model A (L_2 loss). Middle: performance summary of Model B (Sparse Variation loss, $\lambda = 0.2, \rho = 0.1$ in (1)-(3)). Bottom: performance summary of the model ensemble composed of 10 checkpoints sampled during the last 50 epochs of training of model B. Shaded area indicates the $[q_{0.05}, q_{0.95}]$ inter-quantile interval, computed respectively over the last 50 checkpoints for single models, and over a 100 random samples of 10 checkpoints for the bottom plot (ensemble).

vectors for all pixel locations in the feature map and we rather focus on feature vectors corresponding to the locations of aberrations and randomly draw locations in the background to get alternative feature vectors. This bag of feature vector computed on the train dataset is projected onto the 2D plane corresponding to the two first principal components that results from PCA decomposition. By tracking the same locations across several training steps, one can visualize how both the aberration classes and the background are separated during training (Fig. 5). A SVM classifier applied to the embeddings retrieved for a single epoch is used to map regions of the latent space to a specific class, which helps visualizing training dynamics. In Fig. 5, we display the PCA latent space of Unet for 3 different epochs (epochs # 82, # 92, and #100). The red and blue points in the 2D scatterplot represent pixel locations associated to dicentric chromosomes and fragments, respectively. The gray points represent locations in the background. Because the snapshots are captured just before

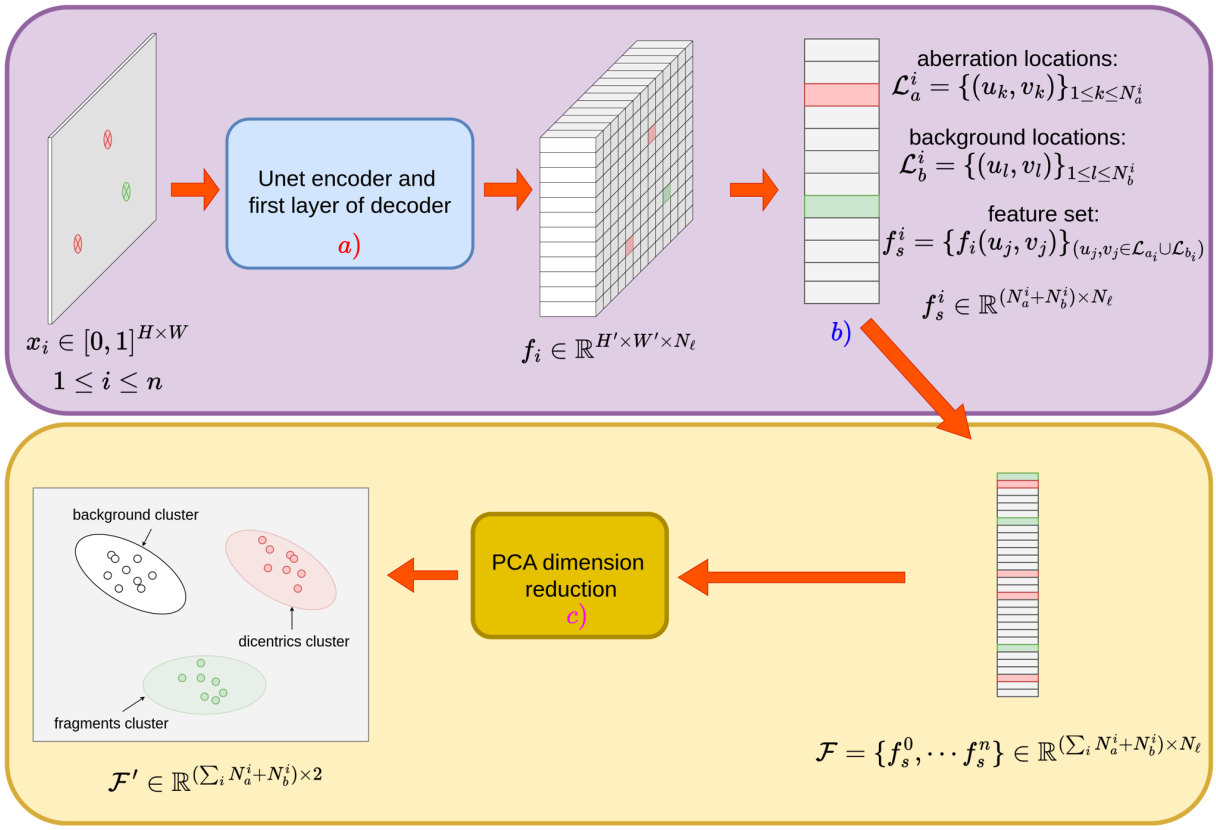


Fig. 4: Method to display feature separation in the latent space of the last decoder block for a single epoch (i.e., a single weight vector θ). For all images $\{x_1, \cdots, x_n\}$, a) the feature maps produced by the last layer of the decoder are computed and b) further consider as a set of independent feature vectors. c) Using PCA dimension reduction, we produce a 2D scatterplot that shows how the model separates the different classes (background, dicentric chromosomes, fragments). The eigenvectors are computed over *all* epochs of training. Because the number of feature vector for each class (background, fragments and dicentrics) is large, we performed K-means clustering on our bag of feature vectors (for each class) and retained only a set of cluster centers for visualization. In the bottom left, each point represents one cluster center.

the end of training, the classes are relatively well separated. One can notice that the decision boundary of the SVM classifier varies at different time points. Regions in the latent space where classes overlap correspond to areas of uncertainty.

While locations with high prediction uncertainty cannot be easily detected because neural networks tend to be overconfident, an ensemble of models can be used to build predictions with accurate confidence estimation. To provide a visualization of this fact, we observe how feature vectors corresponding to image background and each aberration class are separated in feature space during training. We summarize the bag of feature vectors displayed in Fig.5 using K-means by retrieving a small set of centroids. We plot the trajectory of those centroids over training epochs in Fig. 6. It turns out that most most cluster centers are well separated across training epochs. In other words, if we train a SVM classifier on our bag of feature vectors at each epoch and consider the entropy of the average classifier, most centroids will lie in low entropy regions. The centroids lying in high-entropy regions correspond to uncertain detections.

The richness of models is here exploited to compute an ensemble model (i.e., aggregated model) and to estimate uncertainty. Given the set of artificial expert (i.e., models provided from different training ckeckpoints), a decision is made based on a consensus vote which needs to exceed a threshold value. Although the selected checkpoints reach a similar validation performance, the predictions are not similar as illustrated in Fig. 7. We can benefit from these disagreements between checkpoints to discard spurious detections as follows.Each model"votes" for any given object observed in the metaphase image. The model votes for a region of the image if its corresponding heatmap value exceeds a pre-specified confidence threshold. Using a confidence threshold T_C, we build a set of different binary predictions for a given input test image, that we sum over all members of the ensemble at each spatial location in the input image. This leads to a ensemble prediction that takes values between 0 and N (with N being the number of models in the ensemble). Finally, we set an agreement threshold T_A to compute the agreement between the artificial "experts". The setting of thresholds T_A and T_C impact the final decision. If the confidence threshold T_C is high and the voting threshold T_A is low, the decision will be made from a small number of "confident" experts. Otherwise, a small value of T_C but a high voting threshold T_A means that low confidence predictions are considered, but a higher agreement between them is needed to confirm a detection. In the end, we get Precision/Recall surfaces depending on T_C and T_A. Our aggregated decision is a binary image such that value at each spatial location is 0 if no aberration is predicted, and 1 otherwise (see Fig. 8). The agreement threshold T_A can be adjusted to optimize either Precision

or Recall scores, like the confidence threshold T_C.

Precision is a monotonously increasing function of voting and confidence thresholds, while Recall decreases with higher confidence and higher voting thresholds. We do not provide metrics combining Precision and Recall (e.g., F_1 score) as separating those metrics provides a better view of model performance. The balance between both of those scores depends on the final goal (e.g., reduce false positives, false negatives) which can be reached by choosing a specific par (T_C, T_A) of confidence and voting thresholds, respectively. To estimate the sensitivity of Precision and Recall to the sampling of the checkpoints, we evaluate those scores for 100 samples ensembles and we reported the q_{05}, q_{95} interval for Precision and Recall in Fig. 3. This figure shows the lower uncertainty obtained with ensemble of Model B when compared to the uncertainties of single models A and B.

To ensure the readability of ensemble results, we do not report surfaces for Precision and Recall. Instead, we set a single voting threshold and reported the results over all agreement thresholds. Figure 3 provides the model performance as a function of the confidence threshold, which provides insight into the sensitivity of model performance with respect to this threshold. The ensemble provides a significant performance improvement over the single-model baseline. Aggregation does help to discard spurious detections and improves performances. In Tables 1 and 2, we set the parameters (T_C, T_A) to ensure Precision roughly similar between Model B and ensemble Model highlighting a significant gain in Recall both for dicentric chromosomes and fragments.

Overall, there is a large set of threshold combinations that produce large performance improvements over the DCScore baseline. Furthermore, ensembling also reduces performance uncertainty with narrower inter-quartile intervals; the performance is closer between different ensembles than between single checkpoints. This suggests that our results are not dependent on a specific sampling or selection of the ensemble.

2.3 Combining reliable detection of dicentric and fragments yields ionizing dose curves similar to expert curves

All calibration curves of models dedicated to an automated detection of chromosomal aberration tend to overestimate low doses and underestimate high doses in comparison to manual calibration curve. Consequently, we investigated the following strategy to improve performance. First, we studied the Cumulative Distribution Function (CDF) of the maximum heatmap intensity predicted by the members of the ensemble for the dicentric chromosome and fragment class in an independent calibration curve dataset formed by a batch of images exposed at radiation doses ranged from 0 to 4Gy. In order to synchronize the different heatmap intensity levels across the models, we defined the confidence threshold as a certain quantile of the maximal heatmap intensity values of the 4Gy subset images (see Fig. 9). As it is common in biological dosimetry the calibration curves fitted as a linear -quadratic model from the ensemble model counts. The obtained curves for different threshold values are shown in Fig. 10. As our approach also tends to overestimate the low doses and underestimate the high doses compared to a calibration curve

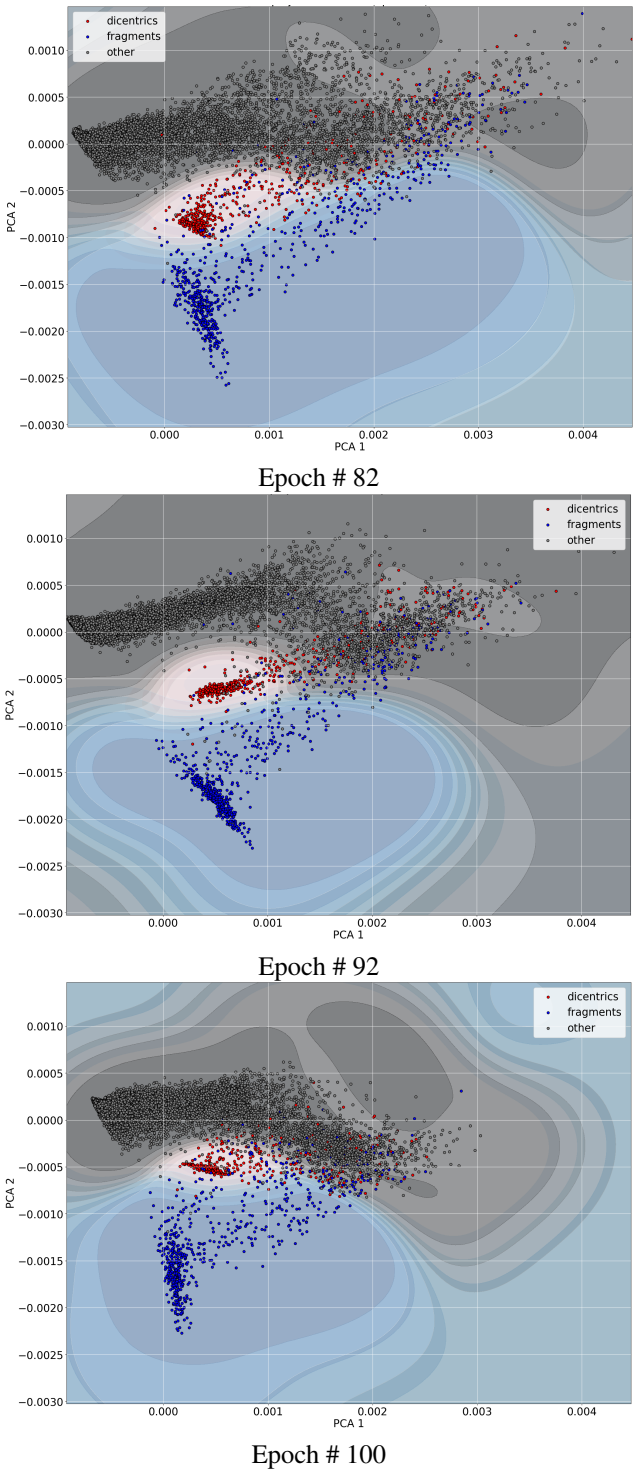


Fig. 5: Snapshot of the training trajectory in feature space. Each point in the three scatterplots at epoch # 82, # 92 and # 100, represents a pixel location in an image with its associated label (dicentric chromosome, fragment or background). Because of the stochasticity of training, the corresponding feature vector moves in feature space. By aligning the feature sets produced by the model at different epochs and computing a PCA dimension reduction, we can visualize the displacement of those feature vectors in feature space during training. The contour map showcases the decision boundary (blue for fragment, white for dicentrics and grey for the background) of a kernel SVM classifier trained to predict the type of feature vector depending on its location in feature space. While some regions of feature space remain ambiguous during training, classes tend to stay clustered together during training.

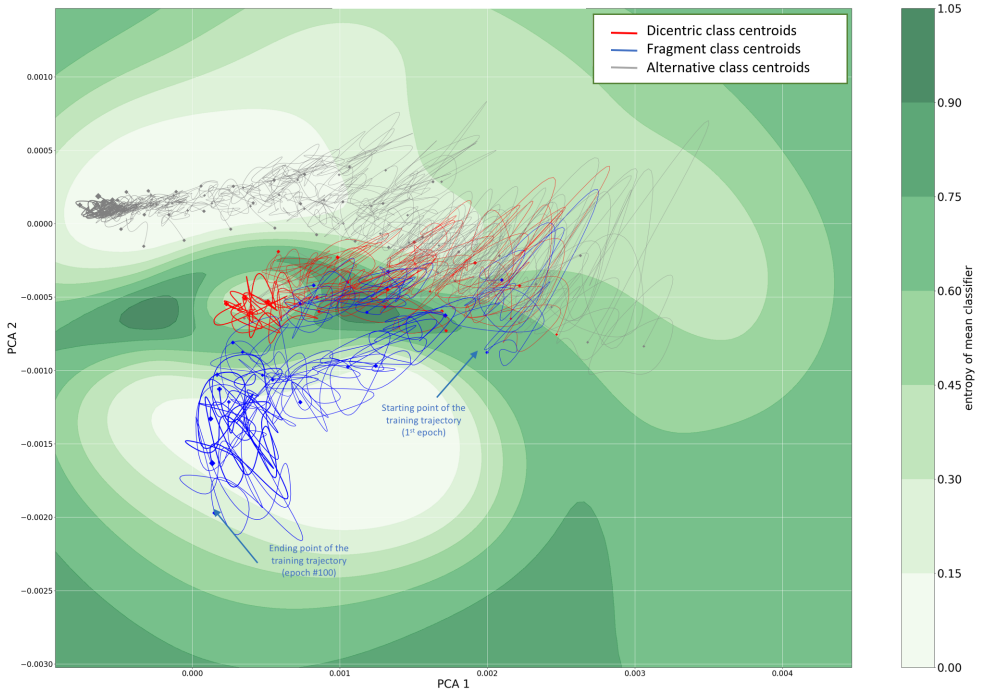


Fig. 6: Training trajectories of feature centroids. As explained in Fig. 4, we summarized our bag of feature vectors by keeping only a set of centroids after K-means clustering in each class. The trajectory of this set of centroids over the last training epochs is displayed above. The thickness of a given trajectory is proportional to the the number of feature vectors contained in the corresponding cluster.

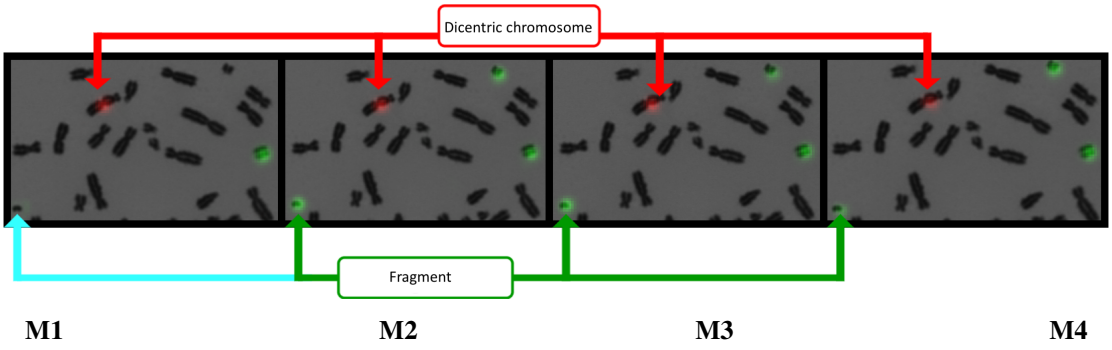


Fig. 7: Prediction diversity for a set of four different models. The models M2, M3, and M4 predict the fragment in the bottom left unlike the model M1. The four models correctly predict the dicentric chromosome in the center of the image.

associated to a manual count, we used prior knowledge and reject spurious dicentric chromosome detection by considering aberration detections if and only if at least one fragment was present in the same image. In Fig. 10, we can assess the significant improvement of the obtained calibration curves and their proximity to the manual curve. Unlike our approach, the DCScore calibration curve is semi-automated, suggesting that dicentric chromosomes undergo a manual review, because of the very high false positive rate of the algorithm.

3 Discussion

In biological dosimetry, estimating the average number of chromosomal aberrations per peripheral blood lymphocyte is mandatory to estimate an ionizing radiation dose. However, human expertise is required and is therefore a bottleneck to scale chromosome counting beyond a few hundred images per patient. To address this issues, we proposed a novel original segmentation CNN-based method for aberration detection that requires no intensive annotation of mono-centric chromosomes, reaches a high level of performance (in terms of Precision and Recall), and provides uncertainty quantification for each detected dicentric chromosome. We consider that the ability of our model to accurately count dicentric chromosomes in unseen test images suggest that our assumptions (only one spot per dicentric chromosome is present, and any overlap between a ground truth and a predicted spot count as a detection) does not lead to misleading estimates of model performance.

In our approach, confidence intervals are computed from models collected during training at the end of each epoch (checkpoint). This can be interpreted as a variant of *thinning*, also used in Monte Carlo Markov Chain inference. To our knowledge, visualizing the latent features of Unet to explore the dynamics during training was not proposed so far.

We showed that the performance uncertainty between differ-

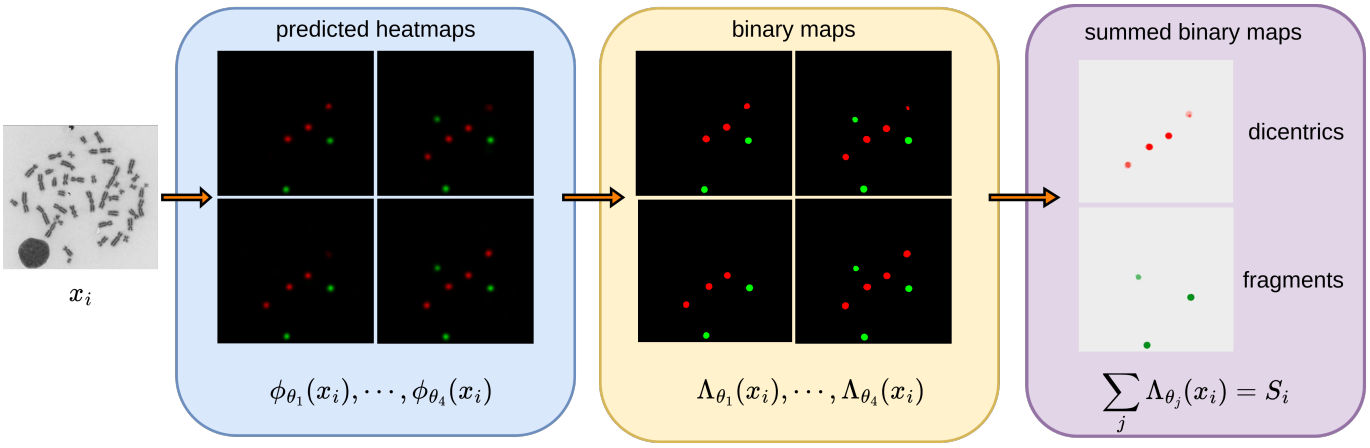


Fig. 8: Vote-based aggregation of four checkpoints. Dicentric chromosomes are plotted in red, and fragment predictions are plotted in green. For each image x_i, the heatmap prediction $\phi_\theta(x_i)$ is binarized to produce the image $\mathbf{\Lambda}_\theta(x_i)$) given a confidence threshold T_C. The binary maps $\mathbf{\Lambda}_\theta(x_i)$ are summed to produce the image S_i, and the pixels of the image getting more than T_A votes are considered as aberration detections. Darker shades of red and green indicates region of the images receiving more votes.

ent (randomly sampled) ensembles is lower than between single checkpoints of a training run. This is especially relevant in the context of the deployment of a deep learning model in an automated fashion in a biomedical setting. Those improvements can be achieved without the need for any architectural modifications or extensive hyperparameter calibration.

Finally, we evaluated our ensemble of Unet in a realistic setting, on a calibration curve dataset. Using model-adaptive thresholding and the domain knowledge of co-occurrence of dicentric chromosomes and fragments, we estimated a very competitive calibration curve, surpassing the DCScore baseline.

4 Methods

Keypoint regression and sparse models. Heatmap regression models assume that objects of interest are represented as Gaussian spots. The model is trained to predict spot positions in the image domain, with a labelled dataset $\mathcal{D} = \{(x_1, y_1), \cdots, (x_n, y_n)\}$ comprised of n realizations of a pair of random variables (X, Y). For each image x_i, we have $x_i(u, v) \in [0, 1]$ at each location $(u, v) \in \Omega$, where Ω denotes the image grid of size $|\Omega| = H \times W$.

Our heatmap regression model is a convolutional neural network $\phi_\theta(x) : x \in [0,1]^{H\times W} \to y \in [0,1]^{H\times W}$. The final output is constrained between 0 and 1 with a sigmoid layer. We use the Unet architecture[26] to predict a low resolution heatmap. The image y_L is of size $(H/L, W/L)$ for some arbitrary downsampling factor L, as the location accuracy provided by the highest resolution output is not useful. For Unet-based architectures, images are usually downsampled (or upsampled, in the decoder) by a factor of 2 at each layer; at layer l of the encoder, features have a spatial size of $H/2^l \times W/2^l$. For the sake of simplicity, notations are given here in the single-channel case. Additional classes of aberrations (e.g., fragments) require an appropriate number of channels and indexes. The parameters θ of our Unet model are learned by solving the following optimization problem:

$$\theta^\star = \arg\min_\theta \sum_{i=1}^{n} \underbrace{\mathcal{L}(\phi_\theta(x_i), y_i) + \lambda\, \mathcal{R}(\phi_\theta(x_i))}_{E_{\theta,\lambda}(x_i, y_i)}, \tag{1}$$

where λ is an hyperparameter that balances the regularization term and the data fidelity term defined as follows:

$$\mathcal{L}(\phi_\theta(x_i), y_i) = \|\phi_\theta(x_i) - y_i\|_2^2. \tag{2}$$

Because the number of aberrations is very low compared to the number of pixels in the image, the background is expected to be 0, except at a few "hot" spots corresponding to aberration locations. The Sparse Variation (SV) regularizer (3)[36] defined as

$$\mathcal{R}(\phi_\theta(x_i)) = \sum_{(u,v)\in\Omega} \sqrt{\rho^2 \|\nabla_{u,v}\phi_\theta(x_i)\|_2^2 + (1-\rho)^2 \phi_\theta(x_i)^2 (u,v)}, \tag{3}$$

has been specifically considered here to encourage the emergence of a very small number of "hot" spots as aberrations are unusual events in Giemsa images. In (3), ρ is a parameter that balances the sparsity and the smoothness terms in the predicted heatmap, and the components of the gradient vector are computed with respect to the image coordinate axes as:

$$\nabla_{u,v}\phi_\theta(x_i) = \begin{bmatrix} \phi_\theta(x_i)(u,v) - \phi_\theta(x_i)(u+1,v) \\ \phi_\theta(x_i)(u,v) - \phi_\theta(x_i)(u,v+1) \end{bmatrix}. \tag{4}$$

The criterion (2) is highly non-convex because of non-linearities in ϕ_θ. Therefore, finding a global minimum is hopeless. Nevertheless, a good local minima may be found using iterative first order methods, such as Stochastic Gradient Descent. The exact gradient of the training criterion with respect to θ is estimated from a random subset $\mathcal{J}$ of the complete dataset $\mathcal{D}$:

$$\nabla_\theta \sum_{i=1}^{n} E_{\theta,\lambda}(x_i, y_i) \simeq \nabla_\theta \sum_{j=1}^{|\mathcal{J}|} E_{\theta,\lambda}(x_j, y_j), \tag{5}$$

to meet GPU memory constraints during training.

Implicit ensembling of neural networks. For each image x_i, $i \in \{1, \cdots, n\}$ in the test set, we consider a set $\{\phi_{\theta_1}(x_i), \cdots, \phi_{\theta_M}(x_i)\}$ of predictions, where M is the number of predictions (or models). Using a confidence threshold T_C, we build a set $\{\mathbf{\Lambda}_{\theta_1}(x_i), \cdots, \mathbf{\Lambda}_{\theta_M}(x_i)\}$ of M different binary predictions for each test image x_i, and we sum over all members of the ensemble at each location $(u, v) \in \Omega$ as follows:

$$\boldsymbol{S}_i(u,v) = \sum_{j=1}^{M} \mathbf{\Lambda}_{\theta_j}(x_i)(u,v), \;\; (u,v) \in \Omega\,. \tag{6}$$

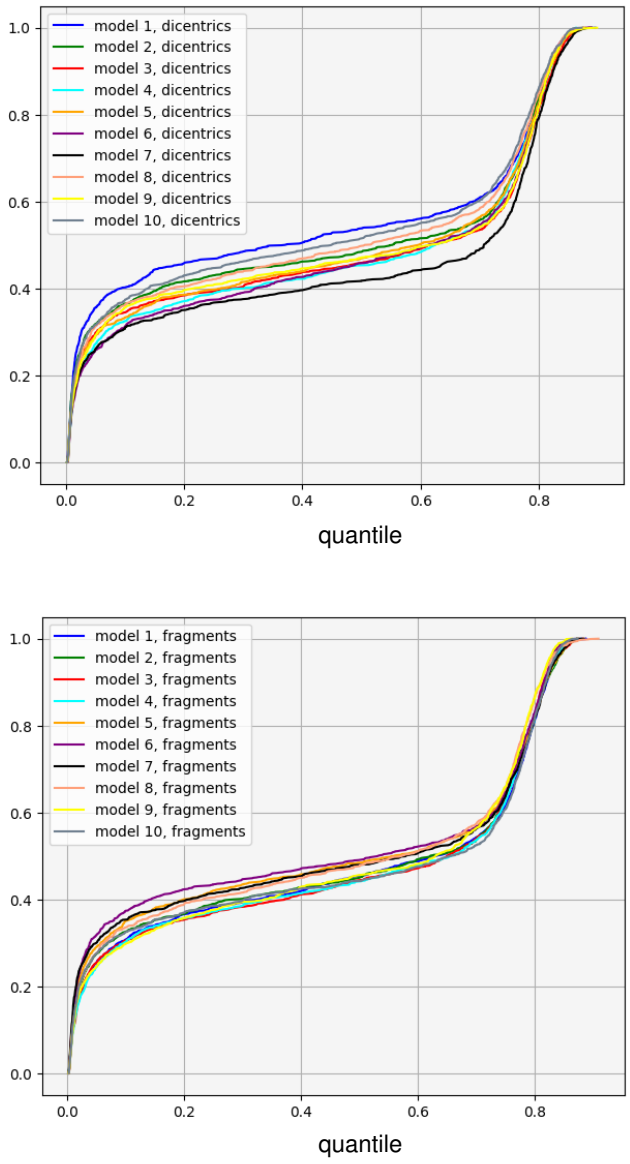


Fig. 9: Quantile distribution analysis. Quantile Distribution Functions (QDF) of the maximum probabilities predicted by each member of the ensemble for the dicentric chromosome class (top) and fragment (bottom) class over all images corresponding to a 4 Gy dose.

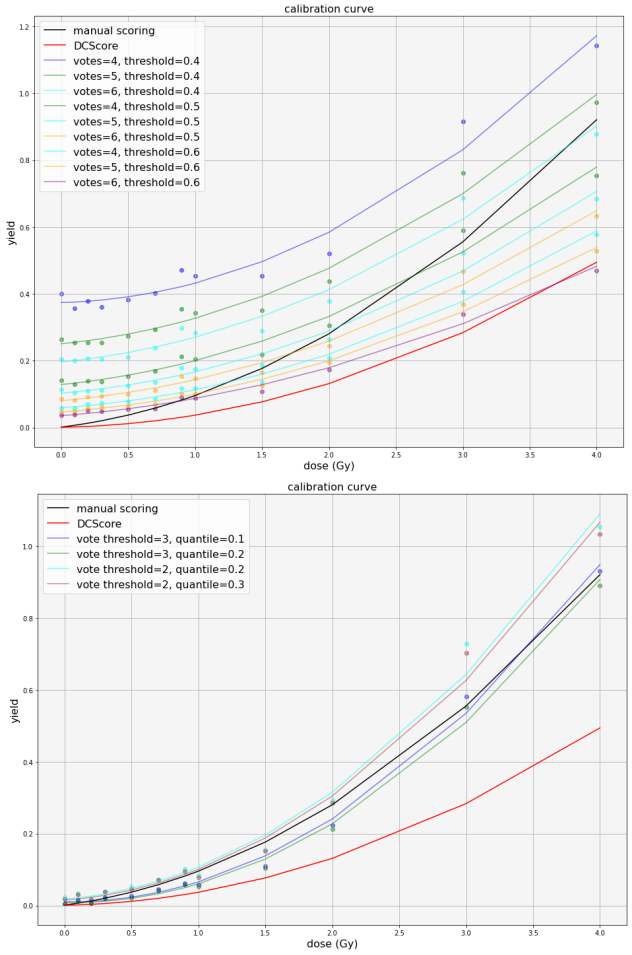


Fig. 10: Calibration curves estimated by the ensemble. Top: calibration curve before setting a threshold per model and using domain knowledge. Bottom: calibration curve after model-adaptive thresholding and using domain knowledge. To improve readability, only the four curves closest to the manual calibration curve (plotted in black) are displayed, while the DCScore curve is plotted in red.

Our aggregated decision for any image x_i is a binary image $\boldsymbol{D}_i$ such that value at location (u, v) is 0 if no aberration is predicted, and 1 otherwise (See Fig. 8 for illustration):

$$\boldsymbol{D}_i(u, v) = \mathbb{1}[\boldsymbol{S}_i(u, v) > T_A], \quad (u, v) \in \Omega. \tag{7}$$

The agreement threshold T_A can be adjusted by the end-user to optimize either Precision or Recall scores, like the confidence threshold T_C.

Setting of model parameters. We trained Unet for $N_e = 100$ epochs epochs with the Adam optimization algorithm [37], with a constant learning rate of 3×10^{-4}, a weight decay parameter of 0.1 and a batch size of 12 on a single Tesla V100. The learning rate was unchanged during training to ensure parameter space exploration, using an analogous reasoning to the one provided in [29].

Data augmentation is usually very effective in improving U-Net performance. It consists in adding noise and blur for instance. However, as the classification of chromosome relies on very small details, we found it difficult to correctly tune data augmentation to avoid performance collapse and therefore choose not to use it. For example, setting the variance of Gaussian blurring slightly too high would blur chromosome too much and make chromosome classification impossible . This is related to our application, and is not true in the general case. We managed to show large improvements over current baselines without data augmentation.

We predict a lower resolution heatmap of size $H' = 224$, $W' = 252$, where height and width are downsampled by a factor of 4. While batch sampling (and therefore parameter space exploration) is randomized, parameter initialization is fixed between training runs. We ran a grid search with $\log_{10}$ spacing for λ regularization parameter with 10 and 10^{-4} as upper and lower bound of the search interval. Training was implemented in PyTorch [38], and uses `segmentation_models_pytorch` implementation of Unet.

Visualization of the training dynamics of single model. Formally, for an input image x_i of size $H \times W$, the ℓ-th layer of our Unet produces a feature volume f_i^ℓ of size $H' \times W' \times N_\ell$ (see Fig. 4 (a)). Here, we choose the second-to-last layer of the decoder and we drop the superscript ℓ to improve readability, so that $f_i^l = f_i$ ($N_l = 128$). We retrieve the set of feature vectors that correspond to the spatial locations of aberrations (dicentric chromosomes and fragments) in image x_i. For a set of

N_a^i aberrations located at $\mathcal{X}_a^i = \{(u_1, v_1), \cdots, (u_{N_a^i}, v_{N_a^i})\}$ in image x_i, we build the following set of feature vectors $f_a^i = \{f_i(u_1, v_1), \cdots, f_i(u_{N_a^i}, v_{N_a^i})\} \in \mathbb{R}^{N_a^i \times N_\ell}$. We retrieve the feature vectors corresponding to all aberrations in each image of the test set. We also sample an additional set of N_b^i background pixels, denoted as f_b^i at locations $\mathcal{X}_b^i = \{(u'_1, v'_1), \cdots, (u'_{N_b^i}, v'_{N_b^i})\}$. Those locations are randomly sampled, provided the locations do not correspond to aberration pixels. Therefore, they correspond to background, monocentric chromosomes or debris. Finally, we define $f_{a,b}^i = f_a^i \cup f_b^i$ so that the total number of feature vectors in f_i is $N_a^i + N_b^i$ (Fig. 4b). It is worth noting that $f_{a,b}^i$ is a subset of the complete feature map f_i. This sparse strategy improves visualizations and reduces computation time. Finally, $f_{a,b}^i$ is retrieved for each image x_i in the test set to build a large feature set $\mathcal{F}_e$ at each epoch e

The set of feature vector $\mathcal{F}_e$ is retrieved for each epoch $e \in \{1, \cdots, N_e\}$. These sets are concatenated in global set $\mathcal{F} = \{\mathcal{F}_1, \cdots, \mathcal{F}_{N_e}\}$. Although all the subsets of $\mathcal{F}$ correspond to the same locations, they are different at each epoch e because of the stochasticity of gradient descent. The PCA-based low dimensionality reduction is used to visualize those feature sets as 2D scatterplots generated from the two first eigenvectors, and in particular to check how well aberrations are separated from the background at each epoch. The set $\mathcal{F}_1$ is chosen as a reference feature set and $\mathcal{F}_e$, $e \in \{2, \cdots, N_e\}$ is registered with respect to $\mathcal{F}_1$ using the Procrustes method[39]. This guarantees that latent space scale shifts or rotations are removed for visualization. A PCA decomposition is computed on $\mathcal{F}$, and for each epoch e, each feature set in $\mathcal{F}_e$ is projected onto the first two principal components of this decomposition. This provides a 2D visualization of the trajectory of feature vectors during training.

Furthermore, we train a kernel SVM classifier p_e on the 2D embeddings of $\mathcal{F}_e$ to predict which aberration class corresponds to a location in 2D embedding space. This classifier takes a 2D vector as an input, and outputs a probability distribution over three classes: background, dicentric chromosome and fragment. For all epochs $1 \leq e \leq N_e$, we train a different classifier, and predict a probability distribution over a grid that samples the 2D aberration space uniformly. For all positions (u, v) of this grid, a probability distribution over the total number N_r of aberration classes $p_e(u, v, r) \in [0, 1]$, $r \in \{0, \cdots, N_r\}$, $\sum_{r=1}^{N_r} p_e(u, v, r) = 1$ is predicted at epoch e. As all the point clouds are aligned and a single set of principal components is computed for all time steps, the changes in the decision boundary from one epoch to the next can be solely attributed to the dynamics of training.

Finally, to visualize the displacement of class boundaries across training, we define the averaged classifier as follows:

$$\bar{p}(u, v) = \frac{1}{N_e} \sum_{e=1}^{N_e} p_e(u, v). \tag{8}$$

Visualizing the spread of the distribution of $\bar{p}$ can be achieved by computing the entropy of the distribution predicted by the averaged classifier:

$$H(\bar{p})(u, v) = -\sum_{r=1}^{N_r} \bar{p}(u, v, r) \log \bar{p}(u, v, r). \tag{9}$$

Dataset description. The images were extracted from a clinical database collected by the Accidental Exposition laboratory at the french Institute for Radiation Protection and Nuclear Safety over the past 3 decades. Those images were then annotated by a set of 5 experts over two months in 2022. As far as we know, there is no publicly dicentric chromosome detection dataset available.

Images have been selected so that each images contains only the chromosomes corresponding to a single cell, i.e., no image contains more than 46 chromosomes. The metaphases in the dataset do not have any missing chromosome, or an excessive chromosome count. Our

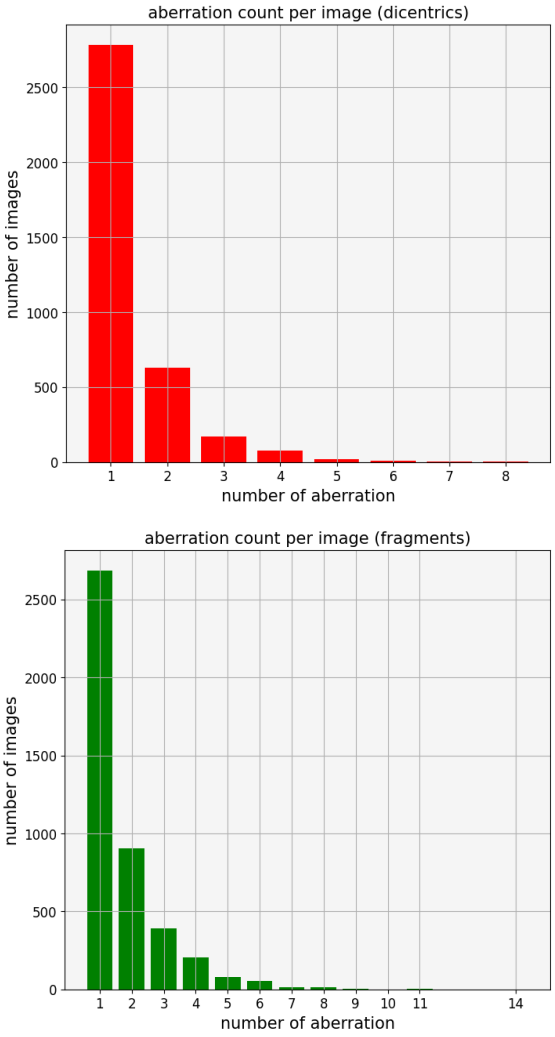


Fig. 11: Distribution of dicentric chromosomes (top) and fragments (bottom) in the metaphase image dataset.

dataset contains 5,021 dicentric chromosomes and 7,540 fragments. Figure **??** displays the empirical probabilities of aberration counts in our dataset. Chromosomal aberrations are the only labelled objects, neither debris nor monocentric chromosomes are labelled. We chose this labelling scheme instead of semantic segmentation or bounding boxes as it leads to the lowest labelling time. This also prevented the discovery of trivial models where chromosomes would be detected but always labelled as monocentric chromosome, as they outnumber dicentric ones by an extremely large margin. On average, there is more than one aberration per image, which corresponds to a very high ionizing radiation dose. As a normal metaphase contains 23 chromosome pairs, this means that even in this case, the overwhelming majority of chromosomes are healthy (i.e., monocentric) ones.

Our selection of training images ensures that metaphase images meet human evaluation standards. This means that chromosomes are spread enough for humans to perform chromosome classification on every one of them. Therefore, we know that there is no overlap of target objects in the training data. This fact is also true for our calibration curve estimation dataset. Our training dataset is then composed of 5,430 labelled images of size $H = 888, W = 1008$, padded to $H = 896, W = 1008$ to ensure that downscaling has an integer height and width. Labels are binary images with size $H' = 202, W' = 252$, taking value 0 everywhere except at the center of chromosomal aberrations (roughly between

the two centromeres for a dicentric chromosome), where it takes value 1. There is one binary image per aberration classes for each image, so that aberration classification is possible. As explained earlier, this binary image is blurred with a Gaussian kernel, to reduce the underrepresentation of the labels against the background. In our evaluation setting, the training set consists of 80% of those image. 10% of the images are retained for a validation set, used for hyperparameter selection. Finally, 10% of the data is held as a test set for a fair performance evaluation.

As this training dataset is not representative of a real-world exposition since it does not contain images without any aberration, we use another dataset to estimate the calibration curve of our model. This dataset was built by collecting metaphases from samples irradiated at specific, known doses. The aberrations in this dataset are not labelled. It contains 21,215 metaphases taken from samples irradiated at 0 Gy, 0.1 Gy, 0.2 Gy, 0.3 Gy, 0.5 Gy, 0.7 Gy, 0.9 Gy, 1.0 Gy, 1.5 Gy, 2.0 Gy, 3.0 Gy, and 4.0 Gy.

Data and software availability. The data cannot be made publicly available due to restricted access under IRSN ethics and security policy, and because informed consent from participants did not cover the publication of this data.

References

[1] AEA, I. Cytogenetic dosimetry: Applications in preparedness for and response to radiation emergencies. Tech. Rep., IAEA - International Atomic Energy Agency (2011).

[2] Bayley, R. et al. Radiation dosimetry by automatic image analysis of dicentric chromosomes. Mutation Research/Environmental Mutagenesis and Related Subjects **253**, 223–235 (1991).

[3] Subasinghe, A. et al. Centromere detection of human metaphase chromosome images using a candidate based method. F1000Research (2016).

[4] Jeong, S. K. et al. Dicentric chromosome assay using a deep learning-based automated system. Scientific Reports **12**, 22097 (2022).

[5] Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks. In Adv. Neural Inf. Processing Systems (NeurIPS), vol. 25 (2012).

[6] Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proc. Med. Image Comput. Comput. Assist. Interv. (MICCAI), vol. 9351 (2015).

[7] Falk, T., Mai, D., Bensch, R. & Ronneberger et al., O. U-net: deep learning for cell counting, detection, and morphometry. Nat. Methods **16**, 67–70 (2019).

[8] Belthangady, C. & Royer, L. Applications, promises, and pitfalls of deep learning for fluorescence image reconstruction. Nat. Methods **16**, 1215–1225 (2019).

[9] Moebel, E. et al. Deep learning improves macromolecule identification in 3d cellular cryo-electron tomograms. Nat. Methods **18**, 1386–1394 (2021).

[10] Wang, J. et al. Chromosome detection in metaphase cell images using morphological priors. IEEE J. Biomedical and Health Informatics **27**, 4579–4590 (2023).

[11] Bai, H. et al. Chromosome extraction based on u-net and yolov3. IEEE Access **8**, 178563–178569 (2020).

[12] Huang, R. et al. A clinical dataset and various baselines for chromosome instance segmentation. IEEE Trans. Comput. Biology and Bionformatics **19**, 31–39 (2022).

[13] Zhang, J. et al. Chromosome classification and straightening based on an interleaved and multi-task ne. IEEE J. Biomedical and Health Informatics **25**, 3240–3251 (2021).

[14] Li, J. et al. Masked conditional variational autoencoders for chromosome straightening. Trans. on Medical Imaging **43**, 216–228 (2024).

[15] Wang, C., Yu, L., Zhu, X., Su, J. & Ma, F. Extended resnet and label feature vector based chromosome classification. IEEE Access **8**, 201098–201108 (2020).

[16] Lin, C. et al. A novel chromosome cluster types identification method using resnext wsl model. Medical Image Analysis **69**, 101943 (2021).

[17] Lin, C. et al. Cir-net: Automatic classification of human chromosome based on inception-resnet architecture. IEEE Trans. Comput. Biology and Bionformatics **19**, 1285–1293 (2022).

[18] Qin, Y. et al. Varifocal-net: a chromosome classification approach using deep convolutional networks. Trans. on Medical Imaging **38**, 2569–2581 (2019).

[19] Xiao, L. et al. Deepacev2: automated chromosome enumeration in metaphase cell images using deep convolutional neural networks. Trans. on Medical Imaging **39**, 3920–3932 (2020).

[20] Uzolas, L., Rico, J., Coupé, P., Sanmiguel, J. & Czerey, G. Deep anomaly generation: An image translation approach of synthesizing abnormal banded chromosome images. IEEE Access **10**, 590090–590098 (2022).

[21] Guo, C., Pleiss, G., Sun, Y. & Weinberger, K. Q. On Calibration of Modern Neural Networks. In Int. Conf. Machine Learning (ICML), 1321–1330 (2017).

[22] Jang, S. et al. Feasibility study on automatic interpretation of radiation dose using deep learning technique for dicentric chromosome assay. Radiation Research **195**, 163–172 (2021).

[23] Shen, X. et al. High-precision automatic identification method for dicentric chromosome images using two-stage convolutional neural network. Scientific Reports **13** (2023).

[24] Jang, S. et al. Radiation dose estimation with multiple artificial neural networks in dicentric chromosome assay. International Journal of Radiation Biology **100**, 865—-874 (2024).

[25] Lambert, B., Frobe, F., Doyle, S., Dehaene, H. & Dojat, M. Trustworthy clinical ai solutions: unified review of uncertainty quantification in deep learning models for medical imaging. Artificial Intelligence in Medicine **150** (2024).

[26] Ronneberger, O., Fischer, P. & Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Proc. Medical Image Computing and Computer-Assisted Intervention (MICCAI), 234–241 (2015).

[27] Vaurijoux, A. et al. Strategy for population triage based on dicentric analysis. Radiation Research **171**, 541–548 (2009).

[28] Lakshminarayanan, B., Pritzel, A. & Blundell, C. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. In Adv. Neural Inf. Processing Systems (NeurIPS), vol. 30 (2017).

[29] Izmailov, P., Podoprikhin, D., Garipov, T., Vetrov, D. & Wilson, A. Averaging weights leads to wider optima and better generalization. In Conference on Uncertainty in Artificial Intelligence (2018).

[30] Maddox, W. J., Izmailov, P., Garipov, T., Vetrov, D. P. & Wilson, A. G. A simple baseline for bayesian uncertainty in deep learning. In Adv. Neural Inf. Processing Systems (NeurIPS), vol. 32 (2019).

[31] Wilson, A. G. & Izmailov, P. Bayesian Deep Learning and a Probabilistic Perspective of Generalization. In Advances in Neural Information Processing Systems (NeurIPS), vol. 33, 4697–4708 (2020).

[32] Shaga Devan, K., Kestler, H. A., Read, C. & Walther, P. Weighted average ensemble-based semantic segmentation in biological electron microscopy images. Histochemistry and Cell Biology **158**, 447–462 (2022).

[33] Huang, G. et al. Snapshot Ensembles: Train 1, Get M for Free. In Int. Conf. Learning Representations (ICLR) (2016).

[34] Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D. P. & Wilson, A. G. Loss Surfaces, Mode Connectivity, and Fast Ensembling of DNNs. In Adv. Neural Inf. Processing Systems (NeurIPS), vol. 31 (2018).

[35] Li, M. & Scheidegger, C. Comparing Deep Neural Nets with UMAP Tour (2021).

[36] Prigent, S. et al. SPITFIR(e): a supermaneuverable algorithm for fast denoising and deconvolution of 3D fluorescence microscopy images and videos. Scientific Reports **13**, 1489 (2023).

[37] Diederik Kingma, J. B. Adam: A Method for Stochastic Optimization. In

Int. Conf. Learning Representations (ICLR) (2015).

[38] Paszke, A. et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Adv. Neural Inf. Processing Systems (NeurIPS), vol. 32 (2019).

[39] Gower, J. C. Generalized procrustes analysis. Psychometrika **40**, 33–51 (1975).

[40] Gruel, G. et al. Biological dosimetry by automated dicentric scoring in a simulated emergency. Radiation rResearch **179** (2013).

Acknowledgements

This work was supported by the Defense Innovation Agency (AID) and the National Research Agency (Increased ANR-20-ASTR-0005, France-BioImaging ANR-10-INBS-04-07). Also, this research was funded in part by IRSN and Région Bretagne.

Author contributions M.A.B and C.K. devised the project and the main conceptual ideas, supervised the project and was in charge of overall direction and planning. A.D. designed and implemented the method, in discussion with E.G, J.S.M., A.V., P.F. D.F., L.B. M.V., and G.G. who designed and provided the datasets. A.D. implemented the method and performed experiments on real images, in discussion with E.M. A.D, C.K., A.M.B., and E.M. co-wrote the manuscript. All authors provided critical feedback and helped shape the research, analysis and manuscript.

Competing Interests

The authors declare no competing interests.