JCBI
Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606
Volume 11 Issue 02
2026

Research Article
https://doi.org/10.56979/1102/2026/1600

Research on the Application of Computer Vision in the Baijiu Distillation Process

Yigao Zhao[1], Yuan Tian[1], Bin Dai[1], Yan Xu[2], and Yanbo Cheng[1*]

[1]Moutai Liquor Co., Ltd, Kweichow, China.
[2]ZTE Corporation, Shenzhen, China.
[*]Corresponding Author: Yanbo Cheng. Email: 0108429@moutai.com.cn

Received: June 03, 2026 **Accepted:** August 16, 2026

Abstract: Background: Bubbles that form in the spirit collector during Baijiu distillation change with alcohol content, and a recent benchmark reported high stage-recognition accuracy with convolutional neural networks. A public repository, the Chinese Spirits Bubble Datasets, hosts representative images and training-loss figures rather than raw images, and the visual information that separates distillation stages has not been audited. **Methods:** We audited the repository and analysed its two reproducible components: 15 detail patches and five overview panels, described by ten interpretable bubble-morphology descriptors, and 14 training-loss figures comparing Global Image with Bubble ROI input, digitised with a validated colour-mask tracer. Stage differences were tested with permutation Kruskal–Wallis tests (Holm correction), stage recognition with leave-one-patch-out classification, and the input comparison with paired Wilcoxon tests and Monte Carlo propagation of digitisation error. **Results:** Seven of the 10 descriptors differed between stages after correction (ε^2 = 0.39–0.93 across descriptors), and a nearest-centroid classifier recognised 14 of 15 held-out patches (accuracy 0.933, 95% CI 0.681–0.998; permutation p = 0.0001; chance 0.200). The patch-trained model did not transfer to overview panels (1 of 5 correct). Across 14 architectures, training loss with Bubble ROI input was higher than with Global Image input in 13 (median θ = 0.67, 95% CI 0.31–0.72; Holm-adjusted exact p = 0.00049), a result that held in all 1,000 digitisation-error draws. **Conclusions:** The representative public images show stage-specific bubble morphology, and Bubble ROI input is associated with slower training-loss reduction. Because test labels are absent, classification accuracy cannot be verified from the repository. **Dataset:** github.com/MengchiCMC/Chinese-Spirits-Bubble-Datasets (commit 6398e32).

Keywords: Computer vision; Baijiu distillation; Bubble morphology; Image-based process monitoring; Convolutional neural networks; Training-loss analysis; Reproducible research; Chinese Spirits Bubble Datasets

1. Introduction

Baijiu is a family of Chinese distilled spirits produced by solid-state fermentation of grains followed by distillation, and the automation of the production chain, from raw-material handling to fermentation and distillation, has recently been reviewed [1]. During distillation the distillate leaves the still in successive fractions whose composition changes over time: according to the documentation of the public Chinese Spirits Bubble Datasets, alcohol by volume decreases and flavour deteriorates as distillation proceeds, so collecting the fractions according to quality is a key step of production [2]. The same documentation notes that spirit quality has traditionally been judged by experienced workers from the visual characteristics of the bubbles in the distillate, and that bubble area, density, colour and size differ between stages [2]. A machine-vision system for classifying Chinese spirits during collection has been built on the observation that the size and shape of the bubbles in the collector change with

alcohol content [3]. Bubble appearance is therefore a plausible, non-contact cue for stage recognition during distillation, and it is attractive because it needs only a camera above the collecting vessel.

Computer vision is now widely used in food processing, where deep learning is increasingly used for classification, grading and defect detection [4, 5]. Applications close to Baijiu production include the detection of impurities in bottled Baijiu, in which air bubbles must be distinguished from foreign particles [6], the grading of Daqu fermentation starter from images [7], and the reading of ethanol concentration from images of an alcoholmeter with a residual network [8]. Closer still to the present topic, bubble size distributions in dense bubbly flows have been detected end-to-end with a YOLO-type detector [9]. A review of machine vision in food detection also cautions that poor lighting, high humidity and image noise can reduce detection accuracy [5].

A closely related benchmark compiled 139,119 distillate images, compared 15 convolutional neural networks (CNNs), and reported accuracies of up to 98.62% with bubble region-of-interest (ROI) selection and 99.82% after a post-processing step based on expert knowledge [3]. The public repository analysed here, the Chinese Spirits Bubble Datasets, was released for researchers who wish to develop visual bubble classification techniques and describes images shot at a large brewery in Sichuan, China [2]. Its README divides distillation into five stages (initial distillate, second distillate, third distillate, last distillate and tail water) and points to an external cloud drive for the image data, but it does not cite a publication. Its content, bubble images of distillation stages and training curves of CNNs under whole-image and ROI input, matches that benchmark, and the correspondence is treated here as an assumption (Section 2.1). What the repository itself contains is smaller: a loader example, two figure files showing representative bubble images, and training-loss figures for the compared networks.

This situation exposes five gaps. First, at the conceptual level, accuracy figures from a deep network do not reveal which visual properties distinguish the stages, and networks can reach high accuracy by exploiting cues that are unrelated to the intended concept [10]. In the sources examined for this study, no interpretable, stage-wise description of the bubble images was found. Second, at the empirical level, the benefit of ROI selection has been summarised by final accuracy, whereas the repository also documents training behaviour for 14 architectures under two input conditions, Global Image and Bubble ROI, which the sources examined do not quantify. Training loss is not a measure of generalisation, since deep networks can fit even random labels [11], so any such comparison needs careful framing. Third, at the methodological level, the repository offers the raw data only through an external link, so quantitative evidence has to be extracted from the figures it holds. Values read from published plots carry digitisation error, and the validity of plot-digitising procedures has to be demonstrated rather than assumed [12]. Fourth, at the contextual level, the repository originates from a single brewery and does not state a brand or aroma type, so claims about other distilleries, or about any named producer, cannot be supported by it. Fifth, at the practical level, reuse of public datasets depends on documentation of content, labelling and licence [13] and on reproducible reporting [14]; no independent audit of what the repository allows a reader to reproduce was found.

The aim of this study is to audit the Chinese Spirits Bubble Datasets repository and to extract, from its reproducible components only, quantitative evidence on stage-specific bubble morphology and on the effect of ROI input on training behaviour. The study pursues three objectives: (i) to document the content, integrity and limits of the repository; (ii) to describe the bubble images of the five distillation stages with interpretable descriptors and test whether those descriptors separate the stages; and (iii) to quantify, with an explicit error model for digitisation, how Bubble ROI input changes the training-loss trajectories of 14 architectures relative to Global Image input.

Three research questions (RQ) follow from these objectives. RQ1: Do interpretable bubble-morphology descriptors differ among the five distillation stages in the representative images of the repository? RQ2: Are these descriptors sufficient to recognise the stage of an unseen patch beyond chance, which descriptor families carry the information, and does the pattern transfer to the lower-magnification overview panels? RQ3: Does the training-loss behaviour of 14 architectures differ between Global Image and Bubble ROI input, and does the difference survive digitisation error? The corresponding hypotheses are as follows. H1: at least some descriptors differ among stages (the null of no stage effect is tested descriptor by descriptor with Holm correction). H2: leave-one-patch-out stage recognition exceeds the chance level of 0.20, and the stage ordering of scale-robust descriptors is

concordant between patches and overview panels. H3: the median log ratio of training loss between the two input conditions differs from zero across architectures. No direction was fixed for H3 in advance.

The study contributes four things. It provides an independent, reproducible audit of what the public repository does and does not contain, including a difference between the 15 networks compared in the related article and the 14 loss-figure pairs in the repository. It supplies interpretable descriptors that differ among the five stages in the representative images and shows where that pattern does not transfer. It quantifies the Global Image versus Bubble ROI difference in training loss across 14 architectures with a validated digitisation procedure and an explicit error model. Finally, it delivers an analysis that fetches the repository automatically and regenerates every number, table and figure reported here. The scope is deliberately bounded: the study does not train new networks, does not re-estimate classification accuracy, and makes no claim about any named producer. The remainder of the paper describes the repository and methods (Section 2), reports the results (Section 3), interprets them and sets out the limitations (Section 4), and closes with the conclusions (Section 5).

2. Materials and Methods

2.1. Dataset and repository audit

The data source is the public Chinese Spirits Bubble Datasets repository on GitHub (https://github.com/MengchiCMC/Chinese-Spirits-Bubble-Datasets), accessed on 7 November 2026 and pinned to commit 6398e32ece2a955d4fef5fe03080fe8758eb5cdc dated 2022-11-28 [2]. The analysis code clones the repository and checks out this commit, so that the same files are used every time it is run. An automatic audit recorded the size, SHA-256 checksum, image dimensions and role of every file, searched for duplicates, parsed the loader script for the record schema and read the README for origin and download information. The audit results are summarised in Table 1.

Table 1. Audit of the Chinese Spirits Bubble Datasets repository (commit 6398e32).

Item	Finding
Repository and commit	MengchiCMC/Chinese-Spirits-Bubble-Datasets, commit 6398e32 (28 November 2022)
Files (excluding .git)	34 files, 30.54 MB
Raw image data (TFRecord)	None; the README links to an external cloud drive
Figure files (PNG)	2: overview (five stage panels) and detail (15 patches, three per stage)
Training-loss files (TIFF)	28 files = 14 unique images (identical pairs in two folders), 600 × 400 px
Networks in the related article	15 CNNs compared; 14 loss-figure pairs in the repository
Licence file	None
Origin stated in the README	“a large brewery in Sichuan, China”; no brand or aroma type

The repository holds 34 files with a total size of 30.54 MB. It contains no TFRecord image files. The loader example expects 224 × 224 × 3 float32 images with an int64 label, which indicates the format of the external image data, but those data are reachable only through a cloud-drive link in the README and were not part of this study. What the repository does contain are (i) two PNG figure files, one with five overview panels (one per stage) and one with 15 detail patches (three per stage); and (ii) 28 TIFF files that reduce to 14 unique training-loss figures, because every figure is stored twice in different folders with identical content. Each loss figure shows two curves of training loss against batch number for one network: a Global Image condition, in which the whole frame is the network input, and a Bubble ROI condition, in which a bubble region of interest is the input. The 14 networks are DenseNet121 [15], EfficientNetB0 and EfficientNetB3 [16], InceptionV3 [17], MobileNet [18], MobileNetV2 [19], MobileNetV3Large and MobileNetV3Small [20], NASNetMobile [21], ResNet50 [22], ShuffleNet [23], SqueezeNet [24], VGG16 [25] and Xception [26]. The related article compared 15 networks [3]; the repository documents 14, and the fifteenth is not identifiable from it. The README does not cite a publication, so the link between the repository and that article is an inference from matching content (bubble ROI input and a comparison of CNNs for spirits classification) and not a documented fact. The repository contains no licence file, and the README gives the origin of the images only as a large brewery in Sichuan, China. The README defines the stages as I (initial distillate),

II (second distillate), III (third distillate), IV (last distillate) and V (tail water) [2]. Because the repository holds neither test labels nor predictions, no classification accuracy could be recomputed, and none is reported as a result of this study.

2.2. Study design and analysis workflow

Given these limits, the study uses two complementary arms built only on what the repository holds (Figure 1). Arm A analyses the representative images: the 15 detail patches and the five overview panels are converted into ten predefined descriptors, which are tested for stage differences (RQ1) and used for leave-one-patch-out stage recognition and a cross-view check on the overview panels (RQ2). Arm B analyses the 14 training-loss figures: the two curves in each figure are digitised, summary metrics are computed, and the two input conditions are compared with paired statistics across architectures, with the digitisation procedure validated on synthetic figures with known truth (RQ3). No network was trained and no image outside the repository was used. The descriptor definitions, the primary outcome of Arm B and its secondary outcome were fixed before any stage-level statistic or loss-curve metric was computed. One candidate descriptor, a Laplacian-of-Gaussian blob scale, was discarded during development because it returned the smallest scale in every patch and therefore carried no image information; it was replaced by the specular-spot descriptors described below.

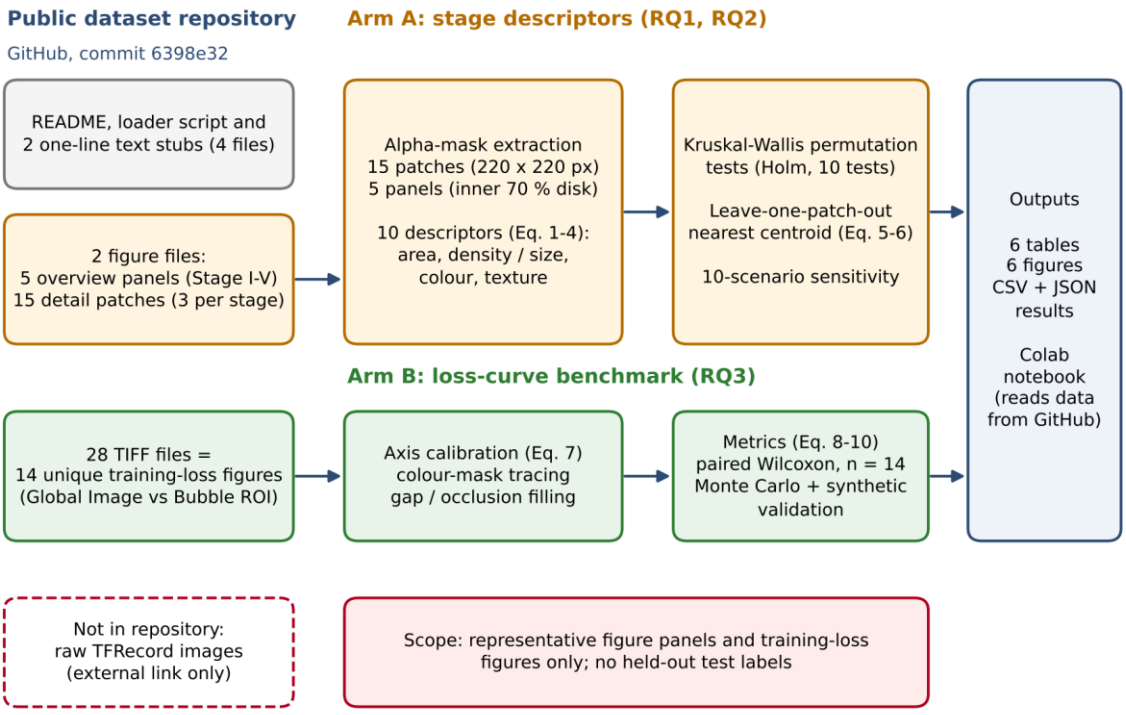


Figure 1. Analysis workflow. The Chinese Spirits Bubble Datasets repository is pinned to commit 6398e32. Arm A (orange) analyses the bubble images; Arm B (green) analyses the training-loss figures; red boxes mark what the repository lacks.

2.3. Image extraction and analysis regions

The detail figure is an RGBA image in which each patch is surrounded by a transparent background. Patches were isolated as connected components of the non-transparent, non-dark pixels after a 5 × 5 morphological closing, keeping components larger than 5,000 pixels. Exactly 15 components were found. They were ordered by row and then by column, so that the three patches of each row form one stage, with the rows corresponding to stages I to V. Each patch was centre-cropped to 220 × 220 pixels (the smallest patch was 223 pixels wide) so that all patches have the same size and contain no background. The overview figure was processed in the same way: five panels were found, ordered from left to right as stages I to V, and the vessel in each panel was filled and reduced to an inner disk with 70% of its equivalent radius to exclude the rim. The stage labels printed in the figures were checked by eye against this order. Figure 2 shows one patch per stage together with the two masks used for the area

and spot descriptors.

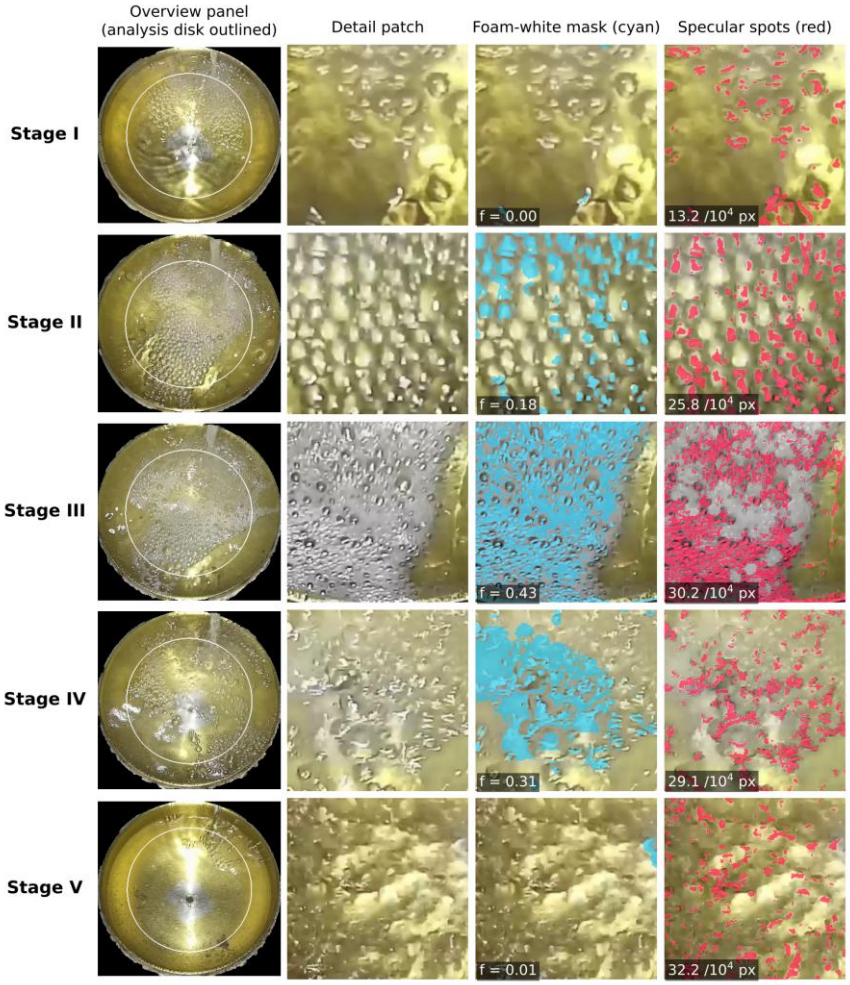


Figure 2. Extraction examples. Rows are stages; columns show the overview panel with the analysed inner disk outlined, the detail patch closest to the stage median foam-white fraction, its foam-white mask (cyan) and its specular-spot mask (red). Inset numbers: foam-white fraction and spots per 10^4 pixels.

The patches and panels are representative images selected by the authors of the repository, and the selection rule is not documented. They are therefore treated as a small, non-random sample of the image population, and inference from them is limited to the images analysed. Whether the three patches of a stage come from one or several bubble sequences is not stated either, so independence of the patches within a stage could not be verified.

2.4. Bubble-morphology descriptors

The README names area, density, colour and size as the visual characteristics of bubbles that differ between stages [2]. Ten descriptors in four families (area, density/size, colour and texture) were defined to cover these characteristics (Table 2). All images were converted from sRGB to CIELAB, giving lightness L*, the coordinates a* and b*, and the chroma $C^*_{ab} = (a^{*2} + b^{*2})^{1/2}$. The foam-white fraction measures the share of analysed pixels that are both light and nearly colourless, as expected for dense white foam:

Table 2. Predefined bubble-morphology descriptors, their families and operational definitions.

Descriptor	Family	Operational definition	Scale-robust
Foam-white fraction (f)	Area	Share of pixels with $L^* \geqslant \tau_L$ and $C^*_{ab} \leqslant \tau_C$ (Eq. 1)	Yes
Specular-spot density (ρ_s)	Density/size	Number of bright connected spots per 10^4 pixels (Eq. 2)	No
Specular-spot median area (A_s)	Density/size	Median area of the bright spots, in pixels (Eq. 2)	No

Descriptor	Family	Operational definition	Scale-robust
Mean L*	Colour	Mean CIELAB lightness	Yes
Mean a*	Colour	Mean CIELAB green – red coordinate	Yes
Mean b*	Colour	Mean CIELAB blue – yellow coordinate	Yes
Edge density	Texture	Share of Canny edge pixels (σ = 2)	No
GLCM contrast	Texture	Mean over four directions, distance 1, 32 levels (Eq. 3)	No
GLCM homogeneity	Texture	Same settings as contrast (Eq. 3)	No
Lightness entropy	Texture	Shannon entropy of the 32-level L* histogram (Eq. 4)	No

$$f = \frac{1}{|\Omega|} \sum_{x \in \Omega} \mathbb{1}\left[L^*(x) \geq \tau_L \ \wedge \ C^*_{ab}(x) \leq \tau_C\right] \tag{1}$$

where Ω is the set of analysed pixels and 1[·] is the indicator function. The thresholds τ_L and τ_C are the Otsu thresholds [27] of the L* and C^*_{ab} histograms pooled over all 15 patches, computed without using the stage labels (τ_L = 68.2, τ_C = 19.0). Specular spots, that is small bright structures such as reflections on bubble surfaces, were detected with a white top-hat transform of L* with a disk of radius 7 pixels, thresholded at the pooled Otsu value τ_T (16.5), and components of four pixels or fewer were removed:

$$S = \{x:\ L^*(x) - (L^* \circ B_7)(x) > \tau_T\}, \qquad \rho_s = 10^4 \frac{N_S}{|\Omega|}, \qquad A_s = \text{median}_k\, |S_k| \tag{2}$$

where B_7 is the disk, ∘ is morphological opening, N_S is the number of connected components S_k of the spot mask, ϱ_s is the spot density and A_s the median spot area. The texture descriptors use the gray-level co-occurrence matrix (GLCM) of L* quantised to 32 levels [28], averaged over the directions 0°, 45°, 90° and 135° at distance 1, with P(i, j) the normalised, symmetric co-occurrence probability:

$$\text{Contrast} = \sum_{i,j} (i-j)^2\, P(i,j), \qquad \text{Homogeneity} = \sum_{i,j} \frac{P(i,j)}{1+|i-j|} \tag{3}$$

Edge density is the share of pixels marked by the Canny detector [29] applied to L*/100 with σ = 2 and hysteresis thresholds of 0.02 and 0.06. Lightness entropy is the Shannon entropy of the 32-level histogram with level probabilities p_k:

$$E = -\sum_{k=1}^{32} p_k \log_2 p_k \tag{4}$$

Four descriptors (foam-white fraction and the three CIELAB means) do not depend on the pixel scale of the image to first order and were designated scale-robust. The overview panels have a lower magnification than the patches, so only these four were computed for the panels, using the same pooled thresholds and the inner vessel disk as the analysis region.

2.5. Statistical analysis

Stage differences (RQ1, H1). Each descriptor was compared across the five stages with the Kruskal–Wallis statistic H [30], corrected for ties, and its effect size was expressed as $\varepsilon^2 = H/(N-1)$ [31], where N = 15 is the number of patches:

$$H = \frac{\frac{12}{N(N+1)} \sum_{g=1}^{5} \frac{R_g^2}{n_g} - 3(N+1)}{1 - \sum_t (t^3 - t)/(N^3 - N)}, \qquad \varepsilon^2 = \frac{H}{N-1} \tag{5}$$

where R_g is the rank sum and n_g = 3 the number of patches of stage g, and the last sum runs over tied groups of size t. With three patches per stage the asymptotic χ^2 approximation is unreliable, so *p*-values were obtained by permuting the stage labels 20,000 times, with $p = (b+1)/(m+1)$ for b permutations at least as extreme in m draws [32]; the smallest attainable value is therefore 0.00005. The ten *p*-values were adjusted with the Holm procedure [33], and a descriptor was called statistically significant when its Holm-adjusted *p* was below 0.05.

Stage recognition (RQ2, H2). Stage recognition was evaluated by leave-one-patch-out (LOPO) cross-validation: in each of the 15 folds one patch was held out, descriptors were standardised with the

mean and standard deviation of the 14 training patches, and the held-out patch was assigned to the stage whose training centroid in standardised descriptor space was nearest (nearest-centroid rule):

$$\hat{y}(x) = \arg\min_{k\in\{I,\ldots,V\}} \| z(x) - c_k \|_2, \qquad c_k = \frac{1}{|T_k|}\sum_{x'\in T_k} z(x') \tag{6}$$

where z(·) is the standardised ten-descriptor vector and T_k the training patches of stage k. The nearest-centroid classifier was the primary model because it has no tunable parameters. A one-nearest-neighbour rule and a logistic regression with L2 penalty (C = 1) [34] served as secondary models. The classification *p*-value is the proportion of 10,000 stage-label permutations (2,000 for the secondary models) whose LOPO accuracy was at least the observed accuracy [35], with the same (b + 1)/(m + 1) correction. Accuracy is reported with the exact Clopper–Pearson 95% confidence interval [36] and the macro-averaged F1 score; the chance level is 0.20. A family ablation repeated the primary analysis with each descriptor family alone and with each family removed.

Cross-view check (RQ2, H2). To see whether the patch-level pattern transfers to the lower-magnification overview panels, the nearest-centroid model was refitted on the four scale-robust descriptors of the 15 patches and applied to the five panels, with the number of correctly assigned panels tested against the binomial distribution with success probability 0.20. In addition, the stage means of each scale-robust descriptor in the patches were correlated with the panel values across the five stages by Spearman's rank correlation, with the exact two-sided *p*-value from all 120 permutations of the ranks. With five points the smallest attainable two-sided *p*-value is 0.0167.

Sensitivity. The robustness of the descriptor pipeline was checked in ten scenarios: the baseline; top-hat radii of 5 and 9 pixels; chroma threshold scaled by 0.8 and 1.2; Canny σ of 1.5 and 2.5; a central 180-pixel crop; Gaussian blurring (σ = 1 pixel); and a resampling scenario in which the image was reduced to 75% of its size and enlarged back by bilinear interpolation. In each scenario the Kruskal–Wallis tests (5,000 permutations) and the LOPO nearest-centroid accuracy were recomputed. The whole Arm A analysis was also repeated with five different random seeds to check that the permutation *p*-values do not depend on the seed.

Precision and resolution. Fifteen patches can be assigned to five labelled groups of three in $15!/(3!)^5$ = 168,168,000 distinct ways, so exhaustive enumeration was impractical and Monte Carlo permutation was used. The estimates remain imprecise: one patch changes the LOPO accuracy by 0.067, and an accuracy of 14/15 has a 95% confidence interval that spans about a third of the unit interval. The sample is fixed by the repository, so no sample-size calculation was possible. The analysis is therefore designed to estimate effect sizes with their uncertainty and not to confirm a pre-specified effect at a target power, and results whose attainable resolution is limited (for example, the correlations over five panels) are labelled exploratory.

2.6. Digitisation of the training-loss figures

Each of the 14 TIFF figures is a 600 × 400 pixel image with the same axis frame (pixel columns 75–540 and rows 48–356) and a fixed colour code: the Global Image curve is blue and dashed, the Bubble ROI curve is orange and dash-dotted. The figures were digitised in four steps. First, the axes were calibrated from the tick marks. The six x ticks (0 to 10,000 batches) and the y ticks (equally spaced from zero to the printed top label of each figure) were located automatically in the image, and a least-squares line mapped pixel coordinates to data coordinates:

$$b = a_x\, u + b_x, \qquad \ell = a_y\, v + b_y \tag{7}$$

where (u, v) are pixel column and row, b is the batch number and ℓ the loss. The top y-axis label of each figure was read from the printed tick labels (values from 1.0 to 20.0). Second, the curve of each condition was extracted as the pixels within an L1 colour distance of 120 from the curve colour and closer to it than to the other curve colour, excluding everything outside the axis frame and the legend box. The vertical position of the curve in each pixel column was the mean row of its pixels. Third, columns without curve pixels were filled. Gaps of up to five columns, which are the gaps between dashes, were interpolated linearly. Where one curve was hidden under the other for longer than five columns and both neighbouring positions agreed with the overlying curve within three pixels, the hidden curve was set equal to the overlying curve. Every column carried a flag for its origin (observed, interpolated or filled from the overlying curve) so that the share of reconstructed columns could be

reported. Fourth, losses closer than two pixels to zero were labelled as below the resolution of the figure, because the pixel grid cannot represent them. This pixel resolution is coarse for figures with a large top label (0.0252 loss units per pixel for ShuffleNet and 0.0725 for VGG16).

2.7. Outcome metrics and paired inference

The primary outcome of Arm B was defined before the metrics were computed. For architecture m and condition c, the time-averaged training loss over the 10,000 batches was obtained by the trapezoidal rule from the digitised curve $\ell_{m,c}(b)$, and the primary outcome θ_m is the log ratio of the Bubble ROI to the Global Image condition:

$$\bar{\ell}_{m,c} = \frac{1}{b_N - b_0}\int_{b_0}^{b_N} \ell_{m,c}(b)\, db, \qquad \theta_m = \ln\frac{\bar{\ell}_{m,\mathrm{ROI}}}{\bar{\ell}_{m,\mathrm{Global}}} \tag{8}$$

A positive θ_m means that the training loss with Bubble ROI input was higher, and $\theta_m = \ln 2 \approx 0.69$ corresponds to a doubling. The log ratio makes architectures with very different loss scales comparable. The secondary outcome is the speed of loss reduction, expressed as the first batch at which the loss falls to 10% of its initial value $\ell(b_0)$, and its paired difference Δt_{90}:

$$t_{90,m,c} = \min\left\{b:\ \ell_{m,c}(b) \le 0.10\,\ell_{m,c}(b_0)\right\}, \qquad \Delta t_{90,m} = t_{90,m,\mathrm{ROI}} - t_{90,m,\mathrm{Global}} \tag{9}$$

The final loss (mean over batches ≥ 9,500) and the initial loss are reported descriptively only, because the final loss is unresolved in most figures. The null hypothesis of no condition effect (H3) was tested on the 14 values of θ_m and $\Delta t_{90,m}$ with the exact two-sided Wilcoxon signed-rank test [37]. With W^+ and W^- the sums of the ranks of the positive and negative differences among the n non-zero values, the effect size is the matched-pairs rank-biserial correlation [38]:

$$r_{rb} = \frac{W^+ - W^-}{n(n+1)/2} \tag{10}$$

The median of θ_m and the median of $\Delta t_{90,m}$ are reported with 95% percentile bootstrap confidence intervals from 10,000 resamples of the 14 architectures [39]; an exact sign test on the count of positive θ_m is added as an assumption-light check. The two *p*-values of the primary and secondary outcomes were adjusted with the Holm procedure. The 14 architectures were trained on the same images, so they are not independent replicates; the test therefore addresses the set of architectures in the repository and not a population of architectures.

2.8. Digitiser validation and error propagation

The digitiser was validated on 50 synthetic figure pairs with known truth. Each pair was drawn with the default matplotlib style, with two curves labelled and coloured as in the repository figures, and generated from two-exponential loss curves with step reductions at multiples of 2,000 batches, which mimics the shape of the repository curves. Three strata were used: separate curves (30 pairs), near-coincident curves that stress the occlusion rule (10 pairs) and curves with a high initial peak, which have a coarse vertical resolution like the figures of ShuffleNet and VGG16 (10 pairs). The digitiser was run unchanged on the rendered images, and the error was expressed as the relative error of the mean loss, the absolute error of θ, the absolute error of t_{90} and the error of the final loss in pixels.

The effect of digitisation error on inference was propagated with 1,000 Monte Carlo draws. In each draw, all positions of a curve were shifted by a uniform offset of ±1 pixel, each column received an additional independent uniform error of ±0.5 pixel, and the x axis was shifted by ±1 pixel:

$$\tilde{v}_j = v_j + u_0 + e_j, \qquad \tilde{u}_j = u_j + w, \qquad u_0 \sim U(-1,1),\ e_j \sim U(-0.5,0.5),\ w \sim U(-1,1) \tag{11}$$

where v_j and u_j are the row and column of curve position j, u_0 and w are the offsets of a draw and e_j is the error of a single column. The metrics, the paired tests and the per-architecture intervals were recomputed for every draw, and the share of draws in which the Wilcoxon *p* stayed below 0.05 was recorded. Two further robustness analyses were pre-planned: excluding the two architectures with coarse vertical resolution (ShuffleNet and VGG16), and leaving out each architecture in turn.

2.9. Software, reproducibility and ethics

All analyses were run in Python 3.13 with NumPy 2.5, SciPy 1.18 [40], scikit-image 0.26 [41], scikit-learn 1.9 [34], pandas 3.0 and matplotlib 3.11, with a fixed random seed (2026). The analysis is

delivered as one notebook that clones the repository from GitHub at the pinned commit, regenerates every table, figure and number, and writes them to a results file; it needs no manual download and no GPU, and runs in a few minutes on a standard CPU runtime. Repeated executions produced identical tables, figures and numbers. The derived data (descriptor tables, digitised curves, calibration tables and validation results), a README and the code are provided as supplementary files, following current recommendations for reproducible machine-learning research and dataset documentation [13, 14]. The study analyses a public repository of spirit-bubble images and training-loss figures and involves no human or animal participants and no personal data; ethical approval was therefore not required.

3. Results

3.1. Repository audit and extraction

The audit confirmed the content summarised in Table 1. The extraction routine found exactly 15 detail patches (minimum width 223 pixels) and five overview panels (widths 370 to 380 pixels) without manual intervention, and every loss figure had the same axis frame, six detectable x ticks and between five and nine detectable y ticks. No raw image, test label or prediction file is part of the repository, so all results below concern the representative images and the training-loss figures. The two arms of the analysis therefore answer RQ1 to RQ3 for these components only.

3.2. Stage differences in bubble morphology (RQ1)

Seven of the ten descriptors differed between the five stages after Holm correction (Table 3; Figure 3), and eight had a permutation p below 0.05 before correction. Effect sizes were large: ε^2 ranged from 0.39 to 0.93 across the ten descriptors, with a median of 0.78. The largest effects were found for GLCM homogeneity ($\varepsilon^2 = 0.93$; Holm-adjusted $p = 0.0005$), specular-spot median area ($\varepsilon^2 = 0.92$; $p = 0.0005$) and edge density ($\varepsilon^2 = 0.92$; $p = 0.0005$). The foam-white fraction also differed between stages ($\varepsilon^2 = 0.87$; permutation $p = 0.0004$; Holm-adjusted $p = 0.0028$), as did spot density ($\varepsilon^2 = 0.75$; $p = 0.0266$), mean b* ($\varepsilon^2 = 0.79$; $p = 0.0192$) and GLCM contrast ($\varepsilon^2 = 0.77$; $p = 0.0232$).

Table 3. Descriptor means by distillation stage (three detail patches per stage) and permutation Kruskal–Wallis tests.

Descriptor	I	II	III	IV	V	*H*	ε^2	*p* (perm.)	*p* (Holm)
Foam-white fraction	0.043	0.178	0.476	0.306	0.020	12.13	0.87	0.0004	0.0028 *
Specular-spot density	12.7	25.3	28.4	23.9	32.3	10.54	0.75	0.0066	0.0266 *
Specular-spot median area	42.7	45.3	11.8	25.3	16.8	12.95	0.92	0.00005	0.0005 *
Mean L*	68.6	66.0	67.1	73.7	65.0	5.43	0.39	0.2623	0.2623
Mean a*	−5.4	−4.8	−0.9	−4.9	−4.3	7.77	0.55	0.0788	0.1576
Mean b*	30.5	20.5	4.2	20.5	29.9	11.03	0.79	0.0032	0.0192 *
Edge density	0.113	0.136	0.198	0.155	0.154	12.83	0.92	0.00005	0.0005 *
GLCM contrast	2.48	3.88	10.31	5.61	3.64	10.77	0.77	0.0046	0.0232 *
GLCM homogeneity	0.752	0.621	0.535	0.657	0.648	13.03	0.93	0.00005	0.0005 *
Lightness entropy	4.04	4.25	3.98	3.73	4.23	8.77	0.63	0.0360	0.1081

Note. H, tie-corrected Kruskal–Wallis statistic (N = 15); permutation p from 20,000 label permutations (minimum 0.00005); Holm correction over ten tests; * Holm-adjusted $p < 0.05$. Spot density per 10^4 pixels; spot area in pixels.

Within-stage variation was considerable. The foam-white fraction of the three patches ranged from 0.003 to 0.122 in stage I, 0.062 to 0.289 in stage II, 0.423 to 0.580 in stage III, 0.277 to 0.331 in stage IV and 0.001 to 0.050 in stage V, so that the ranges of stages I and II, of stages II and IV, and of stages I and V overlapped. The mean b* values of the stage II patches (16.6 to 24.9) likewise covered those of stage IV (19.5 to 22.0). No single descriptor therefore separated all five stages, which is why the combined descriptors are examined in Section 3.3.

Stage III stood out on most descriptors. Its mean foam-white fraction was 0.476, against 0.043 in stage I, 0.178 in stage II, 0.306 in stage IV and 0.020 in stage V. Stage III also had the highest edge density (0.198) and GLCM contrast (10.31), the lowest GLCM homogeneity (0.535), the smallest spots (median area 11.8 pixels) and the lowest mean b* (4.2), which corresponds to a nearly colourless foam. By contrast, stages I and II had large specular spots (median areas 42.7 and 45.3 pixels) and the highest GLCM homogeneity (0.752 in stage I). The spot density rose from 12.7 per 10^4 pixels in stage I to 32.3 in stage V. The profiles were not monotonic along the distillation sequence: stage V resembled stage I in

foam-white fraction (0.020 vs 0.043) and in mean b* (29.9 vs 30.5), and the stage means of L* peaked in stage IV. Three descriptors did not differ after correction: mean L* ($\varepsilon^2 = 0.39$; Holm-adjusted $p = 0.2623$), mean a* ($\varepsilon^2 = 0.55$; $p = 0.1576$) and lightness entropy ($\varepsilon^2 = 0.63$; permutation $p = 0.0360$; Holm-adjusted $p = 0.1081$). Lightness entropy reached nominal significance only before correction and is not counted as a stage difference.

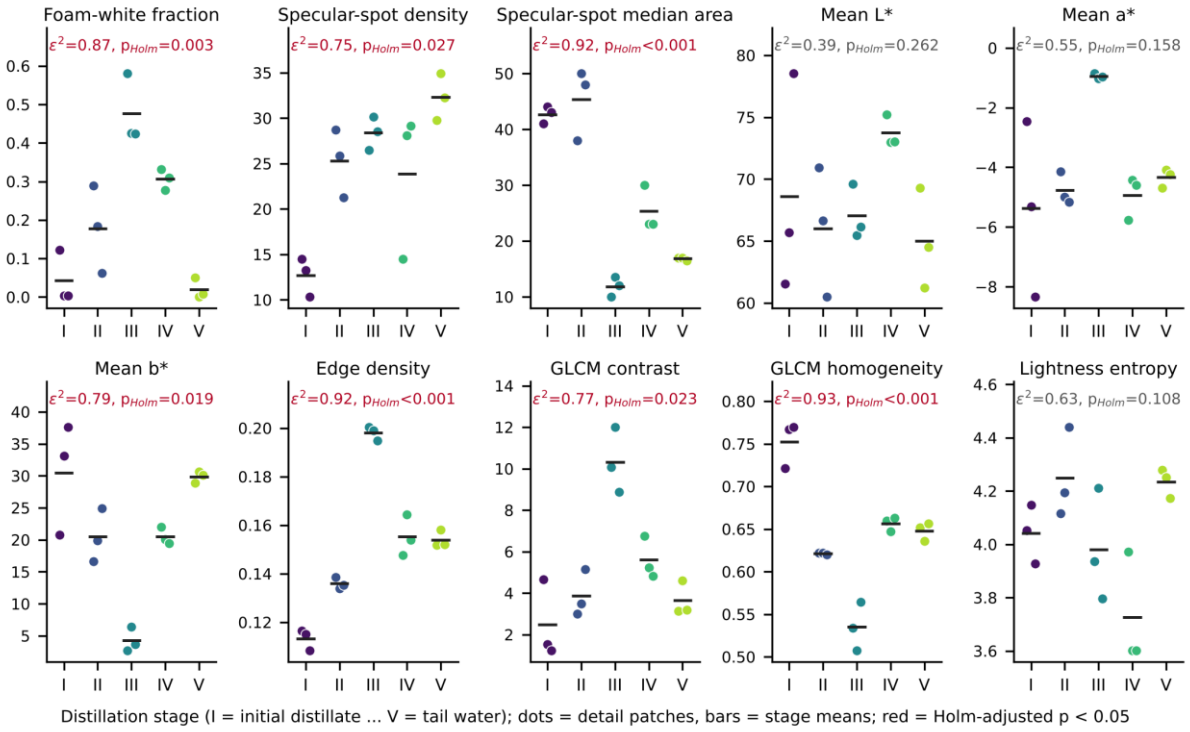


Figure 3. Descriptor profiles by distillation stage. Dots are the three patches of each stage and bars are stage means; panel headings give ε^2 and the Holm-adjusted permutation p of the Kruskal–Wallis test (red, $p < 0.05$).

3.3. Stage recognition and transfer to overview panels (RQ2)

The nearest-centroid classifier recognised 14 of 15 held-out patches (accuracy 0.933; 95% CI 0.681–0.998; macro-F1 0.931), against a mean of 0.180 under 10,000 label permutations and a chance level of 0.200 (permutation $p = 0.0001$, the smallest attainable value; Table 4, Figure 4a,c). The only error was one stage I patch assigned to stage II, so the recall was 0.667 for stage I and 1.000 for each other stage (Figure 4a). The one-nearest-neighbour rule recognised all 15 patches (accuracy 1.000; 95% CI 0.782–1.000; $p = 0.0005$) and logistic regression 12 of 15 (accuracy 0.800; 95% CI 0.519–0.957; macro-F1 0.785; $p = 0.0005$). The wide confidence intervals reflect the small sample, and the three classifiers cannot be ranked from 15 patches. In the first two principal components of the standardised descriptors (59% and 20% of the variance), stage III lies apart along the first and stage IV along the second, whereas stages I, II and V overlap, which is where the single LOPO error occurs (Figure 4b).

Table 4. Stage recognition from the descriptors by leave-one-patch-out (LOPO) cross-validation, family ablation and cross-view transfer to overview panels.

Model or analysis	Accuracy (correct/total)	95% CI	p (vs chance 0.20)
Nearest centroid (primary, 10 descriptors)	0.933 (14/15)	0.681 – 0.998	0.0001
One-nearest neighbour (10 descriptors)	1.000 (15/15)	0.782 – 1.000	0.0005
Logistic regression (10 descriptors)	0.800 (12/15)	0.519 – 0.957	0.0005
Area only	0.600 (9/15)	0.323 – 0.837	–
Density/size only	0.667 (10/15)	0.384 – 0.882	–
Colour only	0.600 (9/15)	0.323 – 0.837	–
Texture only	0.800 (12/15)	0.519 – 0.957	–
Without Area	0.933 (14/15)	0.681 – 0.998	–

Model or analysis	Accuracy (correct/total)	95% CI	*p* (vs chance 0.20)
Without Density/size	0.800 (12/15)	0.519 – 0.957	–
Without Colour	0.933 (14/15)	0.681 – 0.998	–
Without Texture	0.733 (11/15)	0.449 – 0.922	–
Overview panels (cross-view, 4 scale-robust descriptors)	0.200 (1/5)	0.005 – 0.716	0.672 (binomial)

Note. Chance level 0.20. Exact Clopper–Pearson intervals; permutation *p* with 10,000 permutations for the primary model and 2,000 for the others (minimum 0.0001 and 0.0005). The family-only and without-family rows are an ablation without a separate test.

The family ablation indicated that no single family reproduced the accuracy of the full set (Table 4). The texture family alone reached 0.800, the density/size family 0.667, and the area and colour families 0.600 each. Removing the texture family reduced the accuracy most (to 0.733), removing the density/size family reduced it to 0.800, and removing the area or the colour family left it unchanged at 0.933. The pipeline was stable under the ten sensitivity scenarios: between six and seven descriptors stayed significant after correction (six only with a top-hat radius of 9 pixels), the median ε^2 ranged from 0.771 to 0.815, and the LOPO accuracy ranged from 0.867 to 0.933 (baseline 0.933). Repeating the analysis with five random seeds gave seven significant descriptors every time and a maximum difference of 0.0093 in any permutation *p*-value.

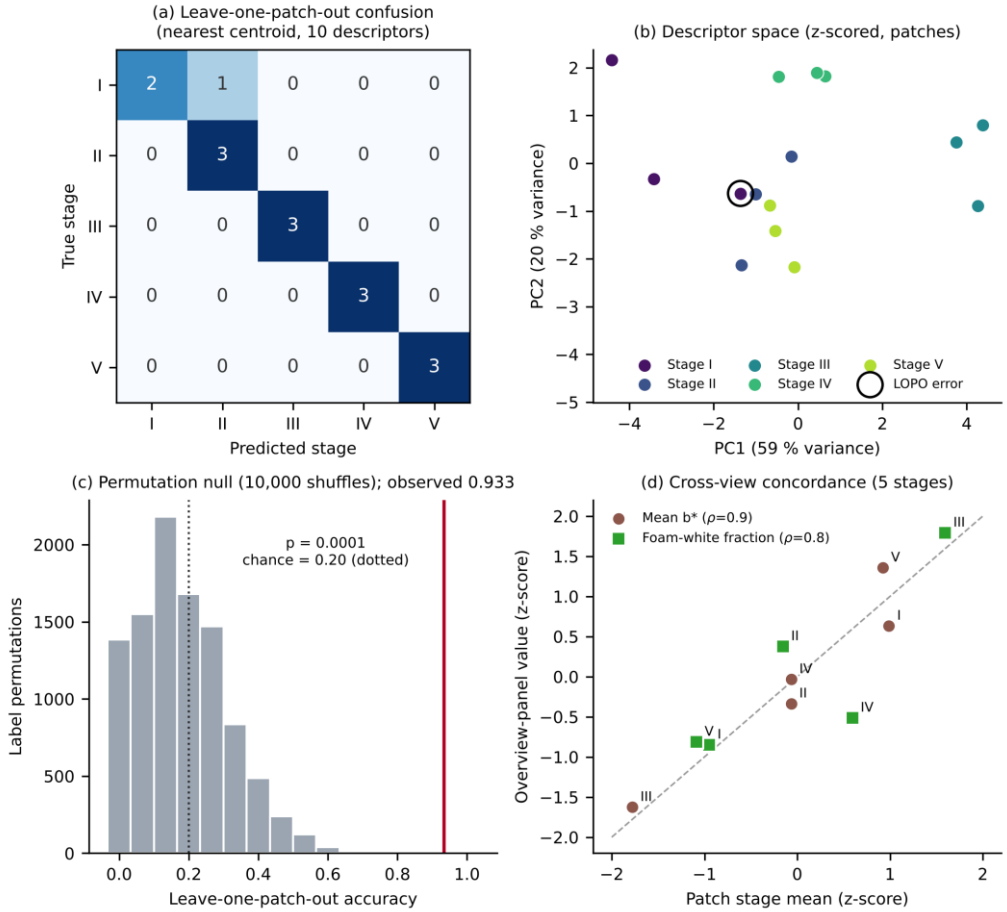


Figure 4. Stage recognition from the descriptors. (a) LOPO confusion matrix of the nearest-centroid classifier; (b) first two principal components of the standardised descriptors, misclassified patch circled; (c) permutation null distribution of the LOPO accuracy (red line, observed; dotted line, chance); (d) cross-view concordance of mean b* and foam-white fraction between patch stage means and overview-panel values.

The patch-trained model did not transfer to the overview panels. Only one of the five panels (stage II)

was assigned to its own stage, a proportion not different from chance (binomial $p = 0.672$); the panels of stages I, III, IV and V were assigned to stages V, II, II and I, respectively. The rank correlations between the patch stage means and the panel values over the five stages were positive but not statistically significant: mean b*, $\varrho = 0.9$ (exact $p = 0.083$); foam-white fraction, $\varrho = 0.8$ ($p = 0.133$); mean L*, $\varrho = 0.4$ ($p = 0.517$); and mean a*, $\varrho = 0.3$ ($p = 0.683$) (Figure 4d). With five stages only a perfect rank agreement could have reached the 0.05 level, so these correlations are compatible with both a weak and a strong relation and are reported as exploratory.

3.4. Digitiser performance and measurement uncertainty

The axis calibration reproduced the printed ticks to within 0.47 pixels on the x axis and 0.58 pixels on the y axis in all 14 figures, and one pixel corresponded on average to 23.7 batches. Across the 28 digitised curves, 63.8% of the pixel columns were observed directly, 29.6% were interpolated across dash gaps of up to five columns, 6.6% were filled from the overlying curve, and no long gap was interpolated. The vertical resolution was coarse for some figures (0.0252 loss units per pixel for ShuffleNet and 0.0725 for VGG16), and the final loss fell below the two-pixel resolution in 12 of 14 figures for the Global Image condition and in 8 of 14 for the Bubble ROI condition. Final losses were therefore not used for inference.

On the 50 synthetic figure pairs, the median absolute relative error of the mean loss was 1.5% (95th percentile 30.5%; maximum 54.7%), and the digitised θ correlated with the true θ at $r = 0.995$ (median absolute error 0.0057; 95th percentile 0.059; maximum 0.208) (Table 5). In the separate and the near-coincident strata, the median error of the mean loss was 1.3% and 1.3%, and the 95th percentile 3.1% and 3.1%, respectively. The error was much larger in the high-peak stratum, which mimics the coarse figures (median 23.8%; 95th percentile 35.5%; maximum 54.7%). The sign of θ was reproduced in 46 of 50 pairs (92%); the four pairs with a wrong sign had true $|\theta|$ of at most 0.034, that is, nearly identical conditions. The median absolute error of t_{90} was 30 batches (95th percentile 123), and the final loss was reproduced within about one pixel (95th percentile 1.01 pixels). The digitiser is therefore accurate for figures with fine vertical resolution and less accurate for figures with a large top label, which motivated the exclusion analysis below.

Table 5. Validation of the digitiser on 50 synthetic figure pairs with known truth.

Stratum	Pairs	Median |error| of mean loss (%)	95th percentile (%)	Median $\|\theta$ error$\|$	Maximum $\|\theta$ error$\|$	Median $\|t_{90}$ error$\|$ (batches)
Separate curves	30	1.3	3.1	0.0050	0.033	37
Near-coincident curves	10	1.3	3.1	0.0048	0.014	60
High initial peak (coarse resolution)	10	23.8	35.5	0.0432	0.208	8
All pairs	50	1.5	30.5	0.0057	0.208	30

Note. Errors are digitised minus true values; θ as in Eq. 8; the high-peak stratum mimics figures with a large top y-axis label.

3.5. Global Image versus Bubble ROI training loss (RQ3)

Training loss with Bubble ROI input was higher than with Global Image input in 13 of the 14 architectures (Table 6; Figure 5). The median θ was 0.67 (95% bootstrap CI 0.31 to 0.72), which corresponds to a median ratio of 1.96 (CI 1.37 to 2.05) between the time-averaged losses. The exact Wilcoxon signed-rank test gave $W = 2$ ($W^+ = 103$, $W^- = 2$), $p = 0.00037$, Holm-adjusted $p = 0.00049$, with a rank-biserial correlation of 0.96; the exact sign test on the 13 positive values gave $p = 0.0018$. The largest values of θ were found for MobileNetV3Small (1.09, about a threefold loss), ResNet50 (0.88) and MobileNet (0.86). VGG16 was the only architecture with a negative θ (−0.15), and its digitisation-error interval (−0.83 to 0.48) included zero, as did that of ShuffleNet (0.06; −0.11 to 0.22). These are the two figures with the coarsest vertical resolution. Three architectures fitted the Bubble ROI input much more slowly than the rest: MobileNetV3Small, MobileNetV3Large and SqueezeNet needed 7,955, 6,602 and 6,414 batches to reach 10% of the initial loss, against 1,997, 2,757 and 2,646 batches with Global Image input, whereas the other eleven architectures reached that level within 1,921 batches in both conditions.

Table 6. Digitised training-loss metrics for the 14 architectures: Global Image versus Bubble ROI input.

Architecture	Mean loss, Global	Mean loss, ROI	θ (95% MC interval)	t_{90} Global	t_{90} ROI	Δ t_{90}
DenseNet121	0.025	0.050	0.69 (0.46 to 0.90)	401	1043	+641
EfficientNetB0	0.042	0.087	0.72 (0.57 to 0.88)	686	1921	+1235
EfficientNetB3	0.055	0.114	0.72 (0.58 to 0.85)	1119	1784	+665
InceptionV3	0.032	0.060	0.64 (0.45 to 0.80)	734	1494	+760
MobileNet	0.037	0.087	0.86 (0.63 to 1.05)	425	1327	+902
MobileNetV2	0.040	0.079	0.68 (0.47 to 0.85)	449	1850	+1401
MobileNetV3Large	0.113	0.171	0.41 (0.36 to 0.46)	2757	6602	+3845
MobileNetV3Small	0.081	0.243	1.09 (1.04 to 1.16)	1997	7955	+5958
NASNetMobile	0.060	0.074	0.22 (0.09 to 0.35)	1000	1736	+736
ResNet50	0.018	0.044	0.88 (0.55 to 1.13)	235	663	+427
ShuffleNet	0.236	0.249	0.06 (−0.11 to 0.22)	905	810	−95
SqueezeNet	0.138	0.180	0.26 (0.23 to 0.30)	2646	6414	+3768
VGG16	0.177	0.152	−0.15 (−0.83 to 0.48)	116	235	+119
Xception	0.027	0.053	0.67 (0.49 to 0.82)	568	1541	+974
Median (95% bootstrap CI)	0.049	0.087	0.67 (0.31 to 0.72)	710	1639	+831 (641 to 1401)

Note. Mean loss is time-averaged over 10,000 batches (Eq. 8); MC interval, 2.5th to 97.5th percentile of 1,000 digitisation-error draws; t_{90}, batches to 10% of the initial loss (Eq. 9); Δt_{90} = ROI − Global. Bottom row: medians with 95% bootstrap CIs.

The Bubble ROI condition was also slower in reducing the loss. The first batch at which the loss fell to 10% of its initial value came later with Bubble ROI input in 13 of 14 architectures, with a median difference Δt_{90} of +831 batches (95% bootstrap CI 641 to 1401; W = 1, exact p = 0.00024; Holm-adjusted p = 0.00049; rank-biserial correlation 0.98). ShuffleNet was the exception (Δt_{90} = −95 batches). The initial loss was higher with Bubble ROI input in 10 of 14 architectures (medians 1.58 vs 1.40). The final loss could be compared in only 2 architectures (MobileNetV3Large and SqueezeNet), where it was higher with Bubble ROI input in both (0.045 vs 0.014 and 0.077 vs 0.025); in the other 12 it was below the figure resolution in at least one condition. Most curves show step-like reductions at multiples of 2,000 batches, which suggests a piecewise-constant learning-rate schedule that the repository does not document.

The result was robust to digitisation error and to the choice of architectures (Figure 6). Across 1,000 Monte Carlo draws the median θ ranged from 0.54 to 0.73 (2.5th to 97.5th percentile), the Wilcoxon p was below 0.05 in 100% of the draws (median 0.00061; 2.5th to 97.5th percentile 0.00012 to 0.0067), and the sign of each architecture's θ was preserved in 95.8% of the draws on average; the median half-width of the per-architecture interval was 0.16. The same held for Δt_{90} (median p = 0.00024; below 0.05 in 100% of the draws). After excluding ShuffleNet and VGG16, all 12 remaining architectures had a positive θ (median 0.68; W = 0; exact p = 0.00049; rank-biserial correlation 1.00), and the median Δt_{90} was +938 batches (exact p = 0.00049). Leaving out each architecture in turn gave exact p-values from 0.00024 to 0.00073 and a median θ from 0.665 to 0.676. The ratio of the summed mean losses, which weights architectures by their loss scale, was 1.52.

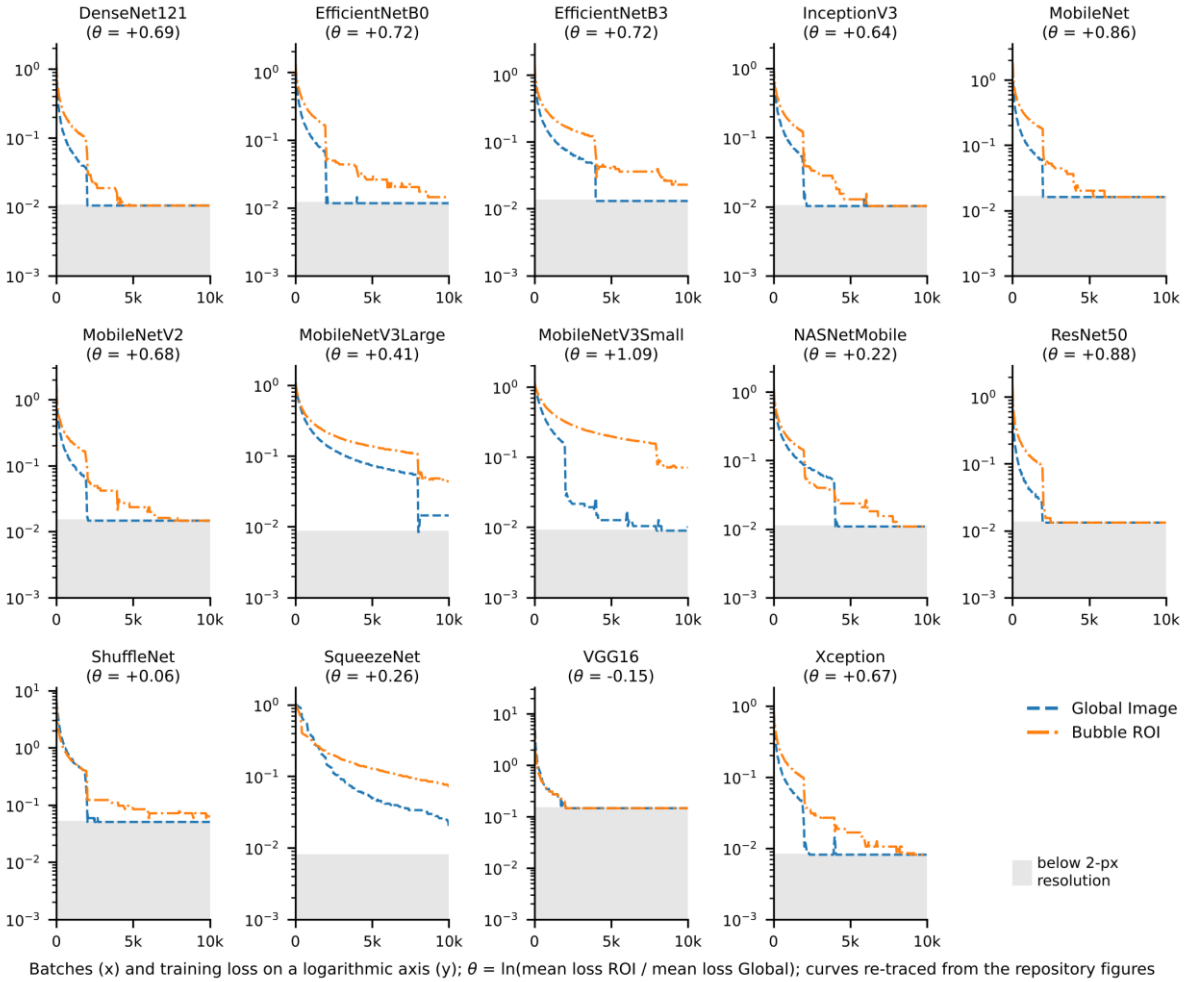


Figure 5. Digitised training-loss trajectories of 14 architectures for Global Image (blue, dashed) and Bubble ROI (orange, dash-dotted) input, on a logarithmic axis. Shaded bands mark losses below the two-pixel resolution, where curves are clipped. Panel titles give θ (Eq. 8).

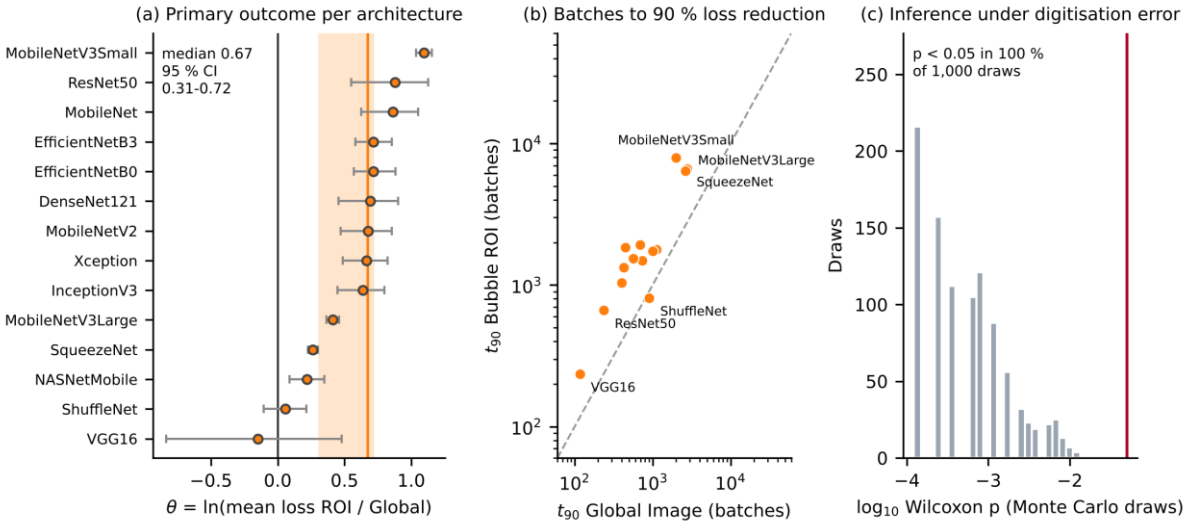


Figure 6. Paired effects of the input condition. (a) θ per architecture with the 95% Monte Carlo interval from digitisation error (orange band, bootstrap CI of the median; line, median); (b) batches to 90% loss reduction, Bubble ROI against Global Image (dashed, equality); (c) Wilcoxon p over 1,000 digitisation-error draws (red line, $p = 0.05$).

4. Discussion

4.1. Principal findings

The three research questions can be answered for the components of the Chinese Spirits Bubble Datasets that are publicly reproducible. For RQ1, seven of the ten descriptors differed between the five distillation stages after correction for multiple testing, with large effect sizes (ε^2 from 0.39 to 0.93 across all descriptors, and from 0.75 to 0.93 among the significant ones). For RQ2, the descriptors recognised 14 of 15 held-out patches (accuracy 0.933; chance 0.200), with the texture family carrying most of the information, but the same descriptors did not recognise the stage of the overview panels (1 of 5), and the rank concordance between patches and panels was not statistically significant. For RQ3, Bubble ROI input was associated with higher time-averaged training loss in 13 of 14 architectures (median ratio 1.96) and with slower loss reduction (median Δt_{90} of +831 batches), and this association survived digitisation error, the removal of the two coarsest figures and the removal of any single architecture. All three hypotheses were therefore supported as far as the evidence in the repository goes, except the second part of H2, the concordance between patches and overview panels, which was not supported.

4.2. Interpretation of the stage morphology

The descriptors were chosen to mirror the characteristics named in the repository README, namely area, density, colour and size [2], and the results show that these characteristics do differ in the representative images, in agreement with that statement. Stage III combined a large foam-white fraction with the highest edge density and GLCM contrast, the lowest homogeneity and the smallest specular spots, which describes a white, dense and finely structured bubble layer. Stages I and II showed larger spots and a smoother texture. The descriptors were computed on a central 220-pixel crop of each patch, so they describe the appearance of the selected region and not the whole bubble layer in the vessel. These are statements about image properties. The repository holds no alcohol measurement or chemical analysis, so the descriptors cannot be tied to alcohol content or to the compounds named in the README, and the study does not test the physical mechanism that links bubble appearance to the composition of the distillate.

Two features of the profiles deserve attention. First, the profiles were not monotonic along the distillation sequence: stage V (tail water) resembled stage I in foam-white fraction and in b*, so a rule based on one of these descriptors alone would confuse the first and the last stage. The family ablation fits this picture, because colour alone (0.600) or area alone (0.600) recognised far fewer patches than the full set, while the texture family alone reached 0.800. Second, the three descriptors that did not differ after correction, L*, a* and lightness entropy, are global intensity statistics, whereas the descriptors that differed describe structure (spots, edges, co-occurrence) or the white-foam proportion. This is in line with the premise that bubble morphology, rather than overall brightness, carries the stage information [3], although brightness is also the descriptor most exposed to illumination changes [5].

The failure of the patch-trained model on the overview panels should not be read as evidence that the panels contain no stage information. The panels have a lower magnification and include large areas of the vessel and the liquid, as Figure 2 shows, and only four scale-robust descriptors could be computed for them. With five panels, a binomial test against chance has almost no power, and a rank correlation can reach significance only with perfect agreement. The positive correlations for b* and the foam-white fraction are compatible with a transfer that is partly present, but they remain exploratory.

4.3. Interpretation of the Global Image versus Bubble ROI comparison

The central finding of Arm B is that models fed with the bubble region reduced the training loss more slowly and had a higher time-averaged loss than models fed with the whole frame. This is a statement about optimisation on the training batches. The related article attributes its highest accuracy to bubble ROI selection [3]; if the Global Image condition of the repository corresponds to its baseline without ROI selection, the models that fit the training data more easily were not the most accurate ones, but the repository does not allow this correspondence to be confirmed. One reading is that the whole frame offers additional cues that are easy to fit but are not the intended bubble cues, such as the vessel, the illumination or camera-specific features, which lower the training loss without improving the decision; this is the pattern described as shortcut learning [10]. A second reading is that the ROI removes redundant context and leaves a harder, more variable input that takes longer to fit. A third is that the two conditions differ in other respects than the ROI, such as the number of pixels, resolution,

augmentation or learning rate, which the repository does not document. None of these readings can be tested with the repository, because it holds no validation or test curves, no masking experiment and no training configuration. The results also do not show that Bubble ROI input generalises better. Deep networks can reach very low training loss on data they cannot generalise from [11], so a lower training loss is neither a sign of a better model nor of a worse one.

Two features of the metrics limit what can be read into the size of the effect. The time-averaged loss is dominated by the early part of training, where the loss is high and the curve is steep, so a condition that starts higher or falls more slowly in the first few thousand batches gains weight in θ. The secondary outcome, t_{90}, is defined relative to each curve's own initial loss, so the slower reduction under Bubble ROI input is not simply a consequence of a higher starting point; the initial loss was in fact higher with Bubble ROI input in only 10 of 14 architectures. The two outcomes are nevertheless not independent measurements, and their agreement should be read as one finding described in two ways.

The result also has two practical uses. It shows that the training loss of a Global Image model cannot be used as a proxy for the training behaviour of the ROI model of the same architecture, since the two differ by a factor of about two on average and by up to a factor of three. It also shows that ROI models needed several hundred more batches to reach the same relative loss reduction, which matters when the training budget is planned.

The present study asks a different question from the related article, which proposed an online classification system, compared 15 CNNs and reported accuracies of up to 98.62% and 99.82% [3]; these could not be reproduced here and were not tested, because the repository holds no test labels. The present results add two things that the repository does not provide: image-level evidence that stage-specific morphology exists in the representative images, which is a precondition for any classifier to work, and a quantitative comparison of the two input conditions during training, which the repository presents only as figure images. Other work on Baijiu has addressed impurity detection in finished liquor [6] and the grading of Daqu starter [7], that is, product inspection and raw-material grading rather than the stage of the distillation. Detector-based approaches that measure bubble size distributions directly [9] would yield richer descriptors than the specular-spot statistics used here, which depend on the pixel scale and on the top-hat radius and are not bubble diameters.

4.4. Implications for monitoring and dataset release

For process monitoring, interpretable descriptors can complement a deep model rather than replace it. A foam-white fraction, a specular-spot size and a texture score can be computed in a few milliseconds from a single frame, can be logged next to the network's stage prediction, and can serve as a plausibility check or as an early warning of drift when illumination or camera position changes, which is a recognised weakness of machine-vision systems in food processing [5]. The present results do not justify using the descriptors as a stand-alone controller, because they were obtained on 15 representative patches from one brewery.

For dataset release, the audit shows what a user needs in order to reproduce and extend a published benchmark. The repository lacks a licence, a mapping from label indices to stages, the raw or test images, the data splits, the training configuration, and validation and test curves, and it holds 14 loss-figure pairs where the related article compares 15 networks. A datasheet in the sense of Gebru et al. [13] and a reproducibility checklist [14] would have closed most of these gaps at a low cost. We therefore suggest that benchmark repositories release at least a labelled test subset with the model predictions, so that the headline accuracy can be recomputed by independent groups.

4.5. Limitations

The study has several limitations that bound its conclusions. First, the sample of Arm A is small and non-random. Fifteen patches, three per stage, were selected by the repository authors by an undocumented rule, and it is not known whether the three patches of a stage come from one bubble sequence. Leave-one-patch-out cross-validation is optimistic if patches from the same sequence are near-duplicates, which would also explain the perfect accuracy of the nearest-neighbour rule. The accuracies reported here are therefore descriptive of the separability of these patches and not estimates of the accuracy on production images, and the confidence intervals are wide. Second, the overview-panel analysis rests on five images and four descriptors. Third, the descriptors and the CIELAB conversion assume that the figure images are faithful sRGB renderings; the images in the

figures may have been resized, compressed or colour-adjusted, and the thresholds are pooled over the 15 patches, so they are not independent of the sample.

Fourth, Arm B depends on digitised figures. The digitiser was validated on synthetic figures that resemble but do not replace the original figures, the final loss could not be resolved for 12 of 14 Global Image figures and 8 of 14 Bubble ROI figures, and two figures had a coarse vertical resolution. The inference on θ and Δt_{90} was shown to be robust to these errors, but the individual per-architecture values carry uncertainty, in particular for ShuffleNet and VGG16. It is also unknown whether the plotted curves are raw or smoothed batch losses. Fifth, the 14 architectures were trained on the same data with a common pipeline, so they are not independent replicates; the paired tests describe these architectures and not architectures in general. Sixth, only training loss is available, so no conclusion on generalisation, accuracy or calibration can be drawn. Seventh, the images come from one brewery in Sichuan whose brand and aroma type are not stated. The study makes no statement about any named producer, about other Baijiu types or about other distilleries, and the title refers to the Baijiu distillation process only in the sense of the repository. Eighth, the repository does not state which publication it accompanies, so its link to the related article is an inference. Ninth, the mapping between the stage labels in the external TFRecord files and the stage names in the README is not documented, and the stage order used here follows the printed labels of the figures. Tenth, the error model of the digitiser assumes independent, bounded and uniformly distributed pixel errors, so systematic errors larger than one pixel are not covered. Finally, the repository carries no licence, so the legal terms of reuse are not specified; only derived measurements are shared with this paper.

4.6. Future work

The most valuable next step is to obtain the external image data and a labelled test split and to repeat the descriptor analysis on the full image population with a cluster-aware cross-validation, so that the sequence-level dependence noted above can be controlled. Replication with images from other distilleries would show whether the descriptor profiles are site-specific. A masking or ROI-ablation experiment, together with validation and test curves, would allow the shortcut-learning hypothesis to be tested directly. More recent architectures, such as vision transformers [42], could be compared under the same two input conditions, and image-based stage estimates could be calibrated against measured alcohol content [8].

5. Conclusions

This study audited the public Chinese Spirits Bubble Datasets repository and analysed its reproducible components with a fully scripted workflow. The repository contains representative bubble images and training-loss figures but neither the raw image data nor test labels. Within these limits, seven of ten interpretable bubble-morphology descriptors differed between the five distillation stages, and the descriptors recognised 14 of 15 held-out patches (accuracy 0.933; permutation $p = 0.0001$), with texture the most informative family. They did not recognise the stage of the five overview panels, so the patch-level pattern cannot be assumed to transfer to lower-magnification views. Across 14 architectures, Bubble ROI input was associated with higher training loss and with slower loss reduction than Global Image input (median $\theta = 0.67$; median $\Delta t_{90} = +831$ batches; Holm-adjusted exact $p = 0.00049$ for both outcomes), and this association persisted under digitisation error.

These findings do not establish classification accuracy, generalisation, or any property of a named producer, because the repository contains no test labels and originates from one brewery of unspecified brand. They show instead what can be checked independently from public materials and where checking stops. Releasing a labelled test subset, a licence, label mappings and training configurations would allow the headline accuracies of image-based distillation monitoring to be verified, and the code and derived data that accompany this paper allow the analysis reported here to be reproduced end to end.

Abbreviations

CNN, convolutional neural network; CI, confidence interval; CIELAB, CIE 1976 L*a*b* colour space; GLCM, gray-level co-occurrence matrix; LOPO, leave-one-patch-out; MC, Monte Carlo; ROI, region of interest; sRGB, standard red-green-blue colour space; TIFF, tagged image file format; θ, log ratio of time-averaged training loss (Bubble ROI/Global Image); t_{90}, batches to 90% loss reduction; ε^2,

epsilon-squared effect size.

Availability of Data and Materials

The analysed data are the public Chinese Spirits Bubble Datasets, available at https://github.com/MengchiCMC/Chinese-Spirits-Bubble-Datasets (commit 6398e32ece2a955d4fef5fe03080fe8758eb5cdc). The raw TFRecord images mentioned in the README are on an external cloud drive and were not used. The analysis notebook, derived data, result tables, figures and a README are provided with this article; the notebook clones the repository at the pinned commit and regenerates every number reported here.

Author Contributions

All authors contributed to the study design, the analysis and interpretation of the results and the writing of the manuscript, and approved the final version.

Ethics Approval and Consent to Participate

Not applicable. The study analysed a public repository of spirit-bubble images and training-loss figures; no human participants, animals or personal data were involved.

Acknowledgment

The authors thank the maintainers of the Chinese Spirits Bubble Datasets repository.

Funding

This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

Conflicts of Interest

The authors declare no conflict of interest.

Declaration of Generative AI Use

The authors contributed to the conceptualization and design of the study, methodology development, data collection and curation, formal analysis, validation, interpretation of results, and manuscript preparation. The authors also participated in visualization, critical review, and editing of the manuscript. All authors read and approved the final version of the manuscript and agree to be accountable for the integrity and accuracy of the work.

References

1. Ye H, Wang J, Shi J, Du J, Zhou Y, Huang M, et al. Automatic and intelligent technologies of solid-state fermentation process of Baijiu production: applications, challenges, and prospects. Foods. 2021;10(3):680. doi:10.3390/foods10030680.
2. MengchiCMC. Chinese-Spirits-Bubble-Datasets [Internet]. GitHub repository, commit 6398e32 (28 November 2022) [accessed 7 November 2026]. Available from: https://github.com/MengchiCMC/Chinese-Spirits-Bubble-Datasets.
3. Chen M, Liu H, Zhang S, Liu Z, Mi J, Huang W, et al. Spirits quality classification based on machine vision technology and expert knowledge. Meas Sci Technol. 2023;34(5):055405. doi:10.1088/1361-6501/acb2e1.
4. Zhu L, Spachos P, Pensini E, Plataniotis KN. Deep learning and machine vision for food processing: a survey. Curr Res Food Sci. 2021;4:233–249.
5. Xiao Z, Wang J, Han L, Guo S, Cui Q. Application of machine vision system in food detection. Front Nutr. 2022;9:888245. doi:10.3389/fnut.2022.888245.
6. Shi X, Tang Z, Wang Y, Xie H, Xu L. HOG-SVM impurity detection method for Chinese liquor (Baijiu) based on adaptive GMM fusion frame difference. Foods. 2022;11(10):1444. doi:10.3390/foods11101444.
7. Zhao M, Han C, Xue T, Ren C, Nie X, Jing X, et al. Establishment of a Daqu grade classification model based on computer vision and machine learning. Foods. 2025;14(4):668. doi:10.3390/foods14040668.
8. Icagic SD, Kvascev GS. A smart alcoholmeter sensor based on deep learning visual perception. Sensors. 2022;22(19):7394. doi:10.3390/s22197394.
9. Chen M, Zhang C, Yang W, Zhang S, Huang W. End-to-end bubble size distribution detection technique in dense bubbly flows based on You Only Look Once architecture. Sensors. 2023;23(14):6582. doi:10.3390/s23146582.
10. Geirhos R, Jacobsen JH, Michaelis C, Zemel R, Brendel W, Bethge M, et al. Shortcut learning in deep neural

networks. Nat Mach Intell. 2020;2(11):665–673. doi:10.1038/s42256-020-00257-z.
11. Zhang C, Bengio S, Hardt M, Recht B, Vinyals O. Understanding deep learning (still) requires rethinking generalization. Commun ACM. 2021;64(3):107–115.
12. Drevon D, Fursa SR, Malcolm AL. Intercoder reliability and validity of WebPlotDigitizer in extracting graphed data. Behav Modif. 2017;41(2):323–339.
13. Gebru T, Morgenstern J, Vecchione B, Vaughan JW, Wallach H, Daumé III H, et al. Datasheets for datasets. Commun ACM. 2021;64(12):86–92.
14. Pineau J, Vincent-Lamarre P, Sinha K, Larivière V, Beygelzimer A, d'Alché-Buc F, et al. Improving reproducibility in machine learning research (a report from the NeurIPS 2019 reproducibility program). J Mach Learn Res. 2021;22(164):1–20.
15. Huang G, Liu Z, van der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017. p. 4700–4708. doi:10.1109/CVPR.2017.243.
16. Tan M, Le Q. EfficientNet: rethinking model scaling for convolutional neural networks. In: Proceedings of the 36th International Conference on Machine Learning (PMLR 97). 2019. p. 6105–6114.
17. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the Inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016. p. 2818–2826.
18. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, et al. MobileNets: efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861. 2017.
19. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen LC. MobileNetV2: inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018. p. 4510–4520.
20. Howard A, Sandler M, Chu G, Chen LC, Chen B, Tan M, et al. Searching for MobileNetV3. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). 2019. p. 1314–1324.
21. Zoph B, Vasudevan V, Shlens J, Le QV. Learning transferable architectures for scalable image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018. p. 8697–8710.
22. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016. p. 770–778.
23. Zhang X, Zhou X, Lin M, Sun J. ShuffleNet: an extremely efficient convolutional neural network for mobile devices. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018. p. 6848–6856.
24. Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ, Keutzer K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. arXiv:1602.07360. 2016.
25. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (ICLR). 2015.
26. Chollet F. Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017. p. 1251–1258. doi:10.1109/CVPR.2017.195.
27. Otsu N. A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern. 1979;9(1):62–66.
28. Haralick RM, Shanmugam K, Dinstein I. Textural features for image classification. IEEE Trans Syst Man Cybern. 1973;SMC-3(6):610–621.
29. Canny J. A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell. 1986;PAMI-8(6):679–698.
30. Kruskal WH, Wallis WA. Use of ranks in one-criterion variance analysis. J Am Stat Assoc. 1952;47(260):583–621.
31. Tomczak M, Tomczak E. The need to report effect size estimates revisited: an overview of some recommended measures of effect size. Trends Sport Sci. 2014;21(1):19–25.
32. Phipson B, Smyth GK. Permutation p-values should never be zero: calculating exact p-values when permutations are randomly drawn. Stat Appl Genet Mol Biol. 2010;9(1):Article 39.
33. Holm S. A simple sequentially rejective multiple test procedure. Scand J Stat. 1979;6(2):65–70.
34. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: machine learning in Python. J Mach Learn Res. 2011;12:2825–2830.
35. Ojala M, Garriga GC. Permutation tests for studying classifier performance. J Mach Learn Res. 2010;11:1833–

1863.
36. Clopper CJ, Pearson ES. The use of confidence or fiducial limits illustrated in the case of the binomial. Biometrika. 1934;26(4):404–413.
37. Wilcoxon F. Individual comparisons by ranking methods. Biometrics Bulletin. 1945;1(6):80–83.
38. Kerby DS. The simple difference formula: an approach to teaching nonparametric correlation. Compr Psychol. 2014;3:11.IT.3.1. doi:10.2466/11.IT.3.1.
39. Efron B. Bootstrap methods: another look at the jackknife. Ann Stat. 1979;7(1):1–26.
40. Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat Methods. 2020;17(3):261–272.
41. van der Walt S, Schönberger JL, Nunez-Iglesias J, Boulogne F, Warner JD, Yager N, et al. scikit-image: image processing in Python. PeerJ. 2014;2:e453.
42. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, et al. An image is worth 16x16 words: transformers for image recognition at scale. In: International Conference on Learning Representations (ICLR). 2021.