Available Online: 23 October 2026 DOI: 10.54254/2977-3903/2026.38187

UAV-assisted cyclist risk assessment based on visual perception and rule-based risk modeling

Ruochong Gu

The High School Affiliated to Renmin University of China, Beijing, China

thomasgu428@163.com

Abstract. Riders of two-wheeled vehicles are among the most vulnerable road users in Chinese cities. Where they are most exposed is hard to judge from a ground-level viewpoint. This paper presents a Unmanned Aerial Vehicle (UAV)-assisted framework for assessing the traffic risk around such riders, combining a YOLO-based object detector with a rule-based risk model. Road users are detected in aerial imagery under a five-class label set. Every neighbouring object then contributes a hazard term that decays with the angle it subtends at the rider. The terms are summed into one interpretable risk index, mapped onto a Low / Medium / High level at percentiles of the observed distribution. The two two-wheeler classes are kept separate during detection and pooled only at the risk-modelling stage, so riders form one group and electric and fuel-powered two-wheelers need not be told apart. Fine-tuned on VisDrone2019-DET, the detector performs well on vehicles, but it recovers fewer than one annotated pedal cyclist in three, the principal constraint on the system. Over the validation split the resulting ordering of riders proves robust, with rank correlations above 0.96 under eight alternative parameter settings, while absolute index values depend on weights set by judgement. A lightweight, interpretable, single-frame pipeline can therefore locate traffic risk points around riders from UAV imagery. Multi-frame tracking and better small-object recall would help it most.

Keywords: UAV imagery, object detection, YOLO, VisDrone, two-wheeled vehicles, cyclist safety, traffic risk assessment, risk hotspot identification

1. Introduction

1.1. Background and motivation

Two-wheeled vehicles have become the dominant mode of short-distance urban mobility in China. The national electric bicycle fleet had reached roughly 350 million units by 2023, and Guangzhou alone generates on the order of 6.85 million e-bike trips per day [1]. Safety has not kept pace. E-bikes were associated with more than half of the traffic injuries and fatalities recorded in that city in 2023 [1]. The problem is structural. Riders are unprotected yet travel at motor-vehicle-like speeds, they share carriageway space with cars, vans, trucks and buses, and they cross repeatedly between motorised and pedestrian space. Field observations of e-bike–vehicle conflicts in Chinese cities confirm that a large share of near-miss events arises from these mixed-traffic interactions rather than from isolated rider error [2].

Copyright: © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

The hazard faced by such a rider comes from the momentary spatial configuration of everything around them: how many road users are nearby, how large they are, and how close they are. Recovering that configuration requires a viewpoint that looks down onto the road surface. From ground level, the very neighbours that define the risk are routinely hidden behind one another. This is the reasoning behind pole-mounted Closed-Circuit Television (CCTV), which remains the most widely deployed source of elevated traffic imagery in Chinese cities.

CCTV is nevertheless a poor data source for an automatic risk-assessment system. Camera poles stand only a few metres above the carriageway, so the view remains strongly oblique and larger vehicles still occlude the two-wheelers beside them. Fixed installations also impose high deployment and maintenance costs and a restricted field of view [3]. Publicly available datasets derived from traffic surveillance are built for other problems, such as vehicle re-identification, licence-plate recognition or single-class vehicle counting. A neighbourhood-based risk model instead needs dense joint annotation of pedestrians, vehicles and two-wheeled vehicles within one frame. Unmanned Aerial Vehicles (UAVs) address both difficulties at once. A drone flying above an intersection yields a near-orthographic view in which a rider and all of its neighbours appear simultaneously and largely without mutual occlusion, and in which image-plane geometry maps directly onto ground-plane geometry. Large annotated UAV benchmarks are also available: VisDrone2019-DET [4] supplies densely labelled aerial imagery of exactly this mixed traffic across fourteen Chinese cities.

1.2. Related work

UAVs have been applied across a broad range of traffic and road-safety tasks, from accident-scene reconstruction to traffic-parameter estimation and trajectory extraction. Outay et al. identify the bird's-eye viewpoint as the principal advantage for recovering lateral movement in heterogeneous traffic with weak lane discipline [5]. Safety-oriented work has largely followed a trajectory-based paradigm. Zong et al. track road users across UAV video frames and compute pairwise time-to-collision values, flagging interactions below a threshold as critical [6]. Chen et al. instead combine drone and roadside footage of e-bike–vehicle conflicts at signalised intersections with random-parameter discrete-choice models, and identify the rider behaviours associated with high-severity conflicts [2]. Both directions require either multi-frame tracking with reliable identity association or a labelled conflict dataset. Neither is available in the present project.

On the perception side, single-stage detectors of the YOLO family [7, 8] have become the default choice for aerial imagery because they combine real-time throughput with acceptable accuracy on small objects, and VisDrone2019-DET [4] provides the annotated data needed to fine-tune them for this domain.

Risk models themselves fall broadly into learning-based and rule-based families. A rule-based formulation [9] is adopted here for two reasons. It is interpretable, so a risk value can be traced back to the specific neighbour that produced it. It also requires no risk-labelled training data, which does not exist for this scenario.

1.3. Objectives

The objective of this work is to estimate, from a single UAV frame, how much traffic risk each detected rider is exposed to. The system detects riders together with nearby pedestrians and vehicles and extracts risk factors from their spatial relationships. These factors are aggregated into a single scalar risk index per rider, which is then mapped onto an interpretable Low / Medium / High level. The intended use is the identification of traffic risk points: locations and configurations at which riders are systematically more exposed, aggregated over a survey flight to support planning, enforcement and infrastructure decisions. An alert to the individual rider is

out of reach, since no plausible channel exists from an airborne observer to a rider in the seconds available. The framework is therefore deployable with exactly the data it is evaluated on, namely offline aerial imagery.

1.4. Paper organization

The remainder of this paper is organized as follows. Section 2 describes the methodology, covering the overall pipeline, the YOLO-based detector and the design of the rider risk index. Section 3 describes the VisDrone dataset, the five-class label set derived from it and the preprocessing procedure. Section 4 presents detection results and risk-assessment case studies. Section 5 concludes and outlines directions for future work.

2. Methodology

2.1. Overall pipeline

The system is a single-frame, feed-forward pipeline. It takes a UAV frame through object detection and risk factor extraction to a risk index for every rider. Figure 2 illustrates its overall effect on one frame, and Section 4.3 discusses that example.

A single aerial frame is passed to a fine-tuned YOLO detector, which returns axis-aligned bounding boxes with class labels and confidence scores (Section 2.2). Each detection is reduced to an anchor point, a characteristic length and a hazard class. The detections belonging to the two two-wheeler classes are identified as the *risk targets*. Every target–neighbour pair is then converted into a scalar hazard contribution by a distance-decaying risk function. The contributions around each target are aggregated into a single risk index and mapped onto a Low / Medium / High level (Section 2.3).

The pipeline is stateless: it consumes one frame at a time and carries no information between frames, so it needs neither multi-object tracking nor identity association. This keeps the method applicable to the still aerial imagery that VisDrone2019-DET provides, and keeps the runtime dominated by the detector alone. Every stage after detection is closed-form and interpretable. A risk value can always be decomposed back into the individual neighbours that produced it, so the output carries its own explanation. The intended use is the identification of traffic risk points, that is, locations at which riders are systematically more exposed, obtained by aggregating indices over a survey flight. An alert to the individual rider lies outside the reach of the framework, since no plausible channel exists from an airborne observer to a rider in the seconds available.

2.2. Object detection

2.2.1. Detector and training configuration

Detection uses a single-stage detector from the Ultralytics YOLO family [7]. Single-stage detectors suit this application well. They process the whole frame in one forward pass, and their anchor-free, multi-scale heads handle the very small instances that dominate aerial imagery. The detector is initialised from COCO-pretrained weights [10] and fine-tuned on the re-mapped VisDrone training split (Section 3). Table 1 is transcribed from the args.yaml that Ultralytics writes into the run directory.

The learning rate and batch size were chosen by a small grid search reported in Section 4.1. Default Ultralytics augmentation (mosaic, random scaling, HSV jitter, horizontal flip) is left unchanged. No aerial-specific augmentation, tiling or test-time augmentation is applied, a deliberate simplification revisited in Section 4.4. Detections are read out at the Ultralytics default confidence threshold of 0.25. This is the only point in the whole pipeline at which the confidence score is used: every box above it enters the risk stage with full weight.

Table 1. Training configuration of the detector

Item	Value
Framework	Ultralytics YOLO
Initialisation	yolo26m.pt (COCO-pretrained)
Input resolution	896 × 896
Epochs (max)	80
Early-stopping patience	25
Batch size	8
Optimiser	AdamW
Initial learning rate lr0	0.0005
Weight decay	0.0005
Mixed precision (AMP)	enabled
Random seed	0
Hardware	1 × NVIDIA A40
Wall-clock training time	≈ 4 h 51 min (17 487 s, 80 epochs)
Best epoch	60 / 80 (mAP50 = 0.629)

YOLO26 is an end-to-end detector that needs no Non-Maximum Suppression (NMS). For the risk stage only, inference uses its class-agnostic head option, followed by an explicit class-agnostic NMS at IoU 0.5. That step removes residual duplicate boxes: the same object detected twice, or detected as both motor and bicycle. Every detection contributes with full weight (Section 2.3), so a duplicate would otherwise add a second full contribution from inside the distance floor. After this step no detected rider has another rider within 0.1 $L(r)$ of its anchor, against 10 of the 6,173 riders in the ground-truth labels. The detection metrics of Section 4.2 come from the standard Ultralytics validator and are unaffected by this step.

2.2.2. Detection classes and the rider set

The detector is trained on the five-class label set defined in Section 3.1: pedestrian, small_vehicle, large_vehicle, bicycle and motor. The two two-wheeler classes are kept separate at the detection stage and pooled only at the risk-modelling stage into the rider set

$$R = \{o \in D : c(o) \in \{bicycle, motor\}\} \tag{1}$$

where D is the set of all detections in the frame and $c(o)$ the class of detection o. Keeping the classes separate during training preserves whatever visual distinction the detector can learn, allows per-class performance to be reported independently in Section 4.2, and leaves room to give the two rider types different risk parameters in future work.

The motor class comprises all motorised two-wheelers. VisDrone does not distinguish electric two-wheelers and mopeds from fuel-powered motorcycles, and this model does not require the distinction. Within R both are treated as riders exposed to the same surrounding traffic, and both receive identical parameters. The dataset's category granularity therefore has no effect on any quantity the model produces, and the results apply to riders of two-wheeled vehicles as a group. This is a statement of scope: how the two populations differ is left to future work (Section 5).

2.3. Rider risk index

2.3.1. Per-detection risk function

Each detection o is reduced to its class $c(o)$, an anchor point $a(o) = (x_o,\ y_o)$ and a characteristic length $L(o)$. For near-nadir aerial imagery the box centre is the appropriate anchor, since the box encloses the ground footprint of the object under this viewpoint. The characteristic length is the geometric mean of the box sides, $L(o) = (w_o\, h_o)^{1/2}$. It is a stable and orientation-insensitive measure of how much ground the object occupies.

The hazard that detection o poses at a location $q = (x_q,\ y_q)$ in the image plane is described by $\rho_o(q)$, the risk function of detection o evaluated at location q. It is the product of a class weight and a distance-decay term:

$$\rho_o(q) = \rho_0 w_{c(o)} \left(\frac{L(o)}{max\{\sqrt{(x_o - x_q)^2 + (y_o - y_q)^2}, D_{min}\}} \right)^2. \tag{2}$$

Here D_{min} is a distance floor. The distance between the anchor $(x_o,\ y_o)$ and q is not allowed to fall below it, which prevents the risk function from diverging when q coincides with, or lies very close to, the anchor. The floor is expressed in units of the rider being assessed, $D_{min} = \delta\, L(r)$ with δ = 0.5, that is, half a rider length (Section 2.3.2). The class weight $w_c(o)$ expresses how much hazard an object of that class poses to a rider. The five values used are 0.60 for pedestrian, 0.80 for bicycle, 1.00 for motor, 1.60 for small_vehicle and 2.40 for large_vehicle. They are ordered by the mass and speed differential each class presents to a rider: a pedestrian is the least dangerous neighbour to be near, a large vehicle the most. These weights are the only quantities in the model set purely by judgement, and Section 4.3 reports how far the ordering of riders survives changes to them.

Under a near-nadir view an object of real size ℓ at real distance d from the rider has apparent size $L \propto \ell / h$, and the image-plane distance between its anchor and q is proportional to d/h, where h is the flight altitude, so

$$\frac{L(o)}{\sqrt{(x_o - x_q)^2 + (y_o - y_q)^2}} = \frac{l}{d} \tag{3}$$

which is the angle the object subtends as seen from q. Its square is proportional to the solid angle, that is, to the fraction of the rider's surroundings that the object occupies. Taking the exponent to be 2 therefore gives Equation (2) a direct reading: each neighbour contributes in proportion to how much of the rider's immediate environment it fills, weighted by how dangerous that class of neighbour is. $S(r)$ below then becomes a class-weighted occupancy of the rider's surroundings. The same identity makes the risk function invariant to image resolution and flight altitude, so one set of thresholds can serve imagery captured at different heights.

No perspective correction is applied. Such a term would raise the weight of objects lower in the frame, on the grounds that an oblique view places them nearer the camera, and it would require the camera geometry. VisDrone supplies no intrinsics, altitude or gimbal angle, so the term would amount to guessing a quantity the data does not contain, and the model leaves it out.

2.3.2. Per-rider index

The risk experienced by a rider r is the sum of the risk functions of its neighbours, each evaluated at the rider's own anchor point:

$$S(r) = \sum_{o \in N(r)} \rho_o(a(r)) \tag{4}$$

where the neighbourhood $N(r)$ is the set of all other detections in the frame whose anchor lies within a distance $\kappa L(r)$ of the rider's anchor; the rider's own contribution is thereby excluded.

The contributions in Equation (4) are summed, so that the index grows with traffic density. A rider surrounded by six vehicles is worse off than a rider beside one identical vehicle, and summation is the

operation that expresses that difference. The neighbourhood radius is measured in multiples of the rider's own characteristic length. A rider plus two-wheeler has an approximately fixed real-world footprint of 1.7–2.0 m, so $L(r)$ acts as a per-image ruler and supplies an approximate metric normalisation without any camera calibration. With $\kappa = 8$ the neighbourhood covers roughly a 15 m radius, about the region another road user could cross within a few seconds at urban speeds. The distance floor $D_{min} = \delta\, L(r)$ of Equation (2), with $\delta = 0.5$, is rider-relative for the same reason.

Because $S(r)$ is an unbounded sum, it is mapped to the unit interval by

$$I(r) = 1 - exp(-S(r)) \in [0, 1). \tag{5}$$

If each neighbour is read as an independent hazard source of intensity ρ, then $I(r)$ is the probability that at least one hazard materialises in a unit interval of exposure. Equation (5) is strictly monotone in S, unlike hard clipping, so two crowded scenes can still be ranked against each other. It is also smooth, so a small change in the scene produces a small change in the index.

The constant ρ_0 is a single positive scale factor applied to every contribution. Its role is to place the resulting distribution of indices on a convenient part of the [0,1) axis, so that the levels have a reproducible scale. The method claims an ordering of riders and a Low / Medium / High assignment, and neither depends on its value. Scaling ρ_0 by any $c > 0$ maps $S(r) \mapsto c\, S(r)$ for every rider. Equation (5) is strictly monotone in S, so the *ranking* of riders is preserved. The thresholds τ_1, τ_2 are percentiles of the empirical distribution of I (Section 2.3.3), so they follow the same monotone map and every rider keeps its level. ρ_0 is therefore fixed by a stated convention: it is set so that the 90th-percentile rider of the validation split receives $I = 0.5$, which places τ_2 at 0.500 by construction. On the validation split this gives $\rho_0 = 0.111$ over the 4,732 detected riders.

For display, the risk functions of all detections are evaluated on a coarse grid over the whole frame and combined by taking the maximum at each grid point. This gives the risk field overlaid in Figure 2(c). A maximum keeps the overlay legible in dense scenes, where a sum saturates. That field is a visual aid and does not enter the index.

2.3.3. Risk levels and threshold selection

Because no ground-truth risk labels exist for aerial traffic imagery, the thresholds τ_1, τ_2 separating Low, Medium and High are set distributionally. The pipeline is run over every image in the validation split, and the index of every detected rider is collected. τ_1 and τ_2 are then taken as the 70th and 90th percentiles of that empirical distribution. On the validation split this gives $\tau_1 = 0.323$ and $\tau_2 = 0.500$, the latter by construction through the normalisation of ρ_0 above, so only τ_1 carries information about the shape of the distribution.

This fixes the prevalence of each label by construction: 70 % Low, 20 % Medium and 10 % High. That is the intended semantics. The index flags riders in an unusually exposed position relative to ordinary urban traffic; it makes no absolute statement about collision probability. Because the risk function is scale-free, the resulting values transfer unchanged to imagery from other altitudes. Two properties follow. The rule is relative, so a uniformly dangerous dataset would still yield 70 % "Low". The 70/90 split is itself a design decision, chosen so that the High band stays small enough to be actionable. Section 4.4 discusses both.

None of the quantities in the risk model is fitted to data. The decay exponent of 2 follows from the geometry of a near-nadir view (Section 2.3.1); $\kappa = 8$ and $\delta = 0.5$ follow from the physical size of a rider; $\rho_0 = 0.111$ is a scale convention; and $\tau_1 = 0.323$ and $\tau_2 = 0.500$ are percentiles of the observed index distribution. The class weights are the exception, since no risk-labelled aerial dataset exists against which they could be fitted. Section 4.3 therefore reports how far the resulting ranking of riders survives changes to the weights and to κ, δ and the decay exponent, and Section 4.4 states the limitation.

3. Data

3.1. VisDrone dataset and class definition

All experiments use VisDrone2019-DET [4], a large-scale aerial object-detection benchmark collected by the AISKYEYE team at Tianjin University. Its detection subset comprises 8,629 static images of up to approximately 2,000 × 1,500 px. They were captured by drone-mounted cameras in fourteen Chinese cities, among them Tianjin, Hong Kong, Guangzhou, Nanjing, Shenyang and Suzhou, and cover urban and suburban scenes at a range of flight altitudes, viewing angles, object densities and lighting conditions. The official partition provides 6,471 training, 548 validation and 1,610 test-dev images. This work fine-tunes on the training split and evaluates on the validation split. Test-dev is not used, because its annotations are withheld for the challenge evaluation server.

Every object is annotated with a bounding box, a category and a score field that acts as a validity flag: a value of 0 marks a region that the benchmark protocol requires to be excluded from both training and evaluation. Ten object categories are evaluated — pedestrian, people, bicycle, car, van, truck, tricycle, awning-tricycle, bus and motor — together with two auxiliary labels, ignored regions (0) and others (11). For this study they are mapped onto the five-class label set of Table 2. Cars and vans are pooled into small_vehicle, and trucks and buses into large_vehicle. The risk model distinguishes neighbouring vehicles by the hazard they present, essentially their mass and physical extent; body style carries no weight in it, and a detector is in any case unlikely to resolve it on small aerial objects. The two two-wheeler categories are carried through unchanged and pooled only at the risk-modelling stage (Section 2.2.2).

Table 2. Mapping from the ten original VisDrone categories to the five-class label set used in this work

YOLO id	Class	VisDrone source category (id)	Role in the risk model
0	pedestrian	pedestrian (1)	Hazard source
1	small_vehicle	car (4), van (5)	Hazard source
2	large_vehicle	truck (6), bus (9)	Hazard source
3	bicycle	bicycle (3)	Protected target and hazard source
4	motor	motor (10)	Protected target and hazard source
—	(not used)	ignored regions (0), people (2), tricycle (7), awning-tricycle (8), others (11)	Discarded during conversion

Three categories are excluded. People (non-walking persons) are excluded because a stationary or seated person is not a moving participant in the traffic configuration the risk model describes. Tricycle and awning-tricycle are excluded to keep the label set small. That is the weaker justification, since motorised delivery tricycles are common in Chinese cities, and the omission is carried forward as a limitation in Section 4.4.

3.2. Data preprocessing

A single conversion script, applied identically to both splits, translates the VisDrone annotations into the YOLO format. Boxes whose score field is 0 are discarded, as the benchmark protocol requires, as are boxes whose category does not appear in Table 2 and degenerate boxes of non-positive size. The remaining boxes are converted from a top-left corner plus width and height to centre coordinates plus width and height, normalised by the image dimensions, and written to one plain-text label file per image. No resizing, tiling, size-based filtering or class balancing is applied. Images are resized to the network resolution during training (Section 2.2.1), and the consequences of not tiling are discussed in Section 4.4.

3.3. Train / validation split

The official partition is used unchanged, so the reported figures remain directly comparable with other work on the same benchmark. The detection subset consists of independent stills rather than consecutive video frames, so there is no risk of the frame-level leakage that affects datasets split randomly over video sequences [3]. After conversion and filtering, the training and validation splits contain 308,065 and 32,057 annotated instances respectively. The distribution is strongly imbalanced. Small vehicles and pedestrians together account for four-fifths of all instances, while bicycle is the rarest class at 10,477 training instances, one-sixteenth of the small_vehicle count. Section 4.2 therefore assesses the two two-wheeler classes on their own terms rather than through an averaged score.

4. Results

4.1. Experimental setup

The detector configuration is given in Table 1. The learning rate and batch size were chosen by a small grid search over $lr_0 \in \{1, 2, 5\} \times 10^{-4}$ and $\{1, 2, 5\} \times 10^{-3}$ at batch size 8, followed by a check at batch size 16 with the learning rate rescaled proportionally. Validation mAP50 spans 0.618–0.629 across that entire range on a single seed, a spread comparable to what a change of random seed would produce. The detector is therefore insensitive to the learning rate over two orders of magnitude, and no particular value is an optimum. The operating point retained is $lr_0 = 5 \times 10^{-4}$ at batch size 8. Doubling the batch size did not raise the plateau, so the smaller and less memory-hungry configuration was kept. Training ran for 80 epochs in 4 h 51 min on a single NVIDIA A40, with the best epoch at 60.

4.2. Object detection performance

Figure 1 shows the training and validation losses together with the validation metrics over the 80 epochs. Both loss curves fall smoothly and the validation metrics plateau well before the end of training, with the best epoch at 60 out of 80 and early stopping not triggered.

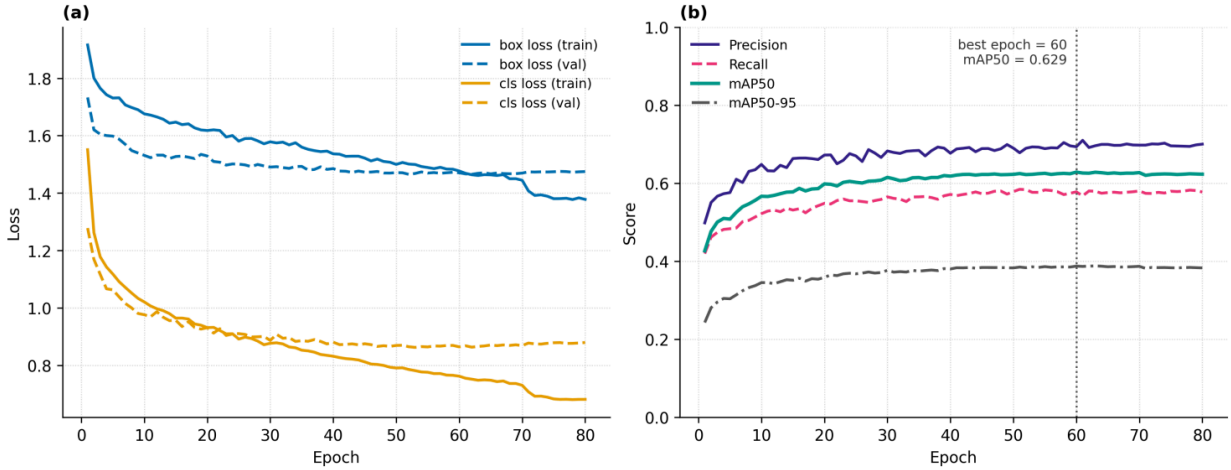


Figure 1. Training of the detector (YOLO26m, 896 × 896 input, $lr_0 = 5 \times 10^{-4}$, 80 epochs). (a) Box and classification losses per epoch, for the training split (solid) and the validation split (dashed). (b) Validation precision, recall, mAP50 and mAP50-95 per epoch; the dotted vertical line marks the best epoch, 60 (mAP50 = 0.629). Early stopping was not triggered

On the 548-image validation split the detector reaches mAP50 = 0.628 averaged over the five classes, with precision 0.693, recall 0.581 and mAP50-95 = 0.389, as computed by the Ultralytics validator. Inference costs 37.6 ms per image at 896 × 896 (2.6 ms preprocessing, 34.7 ms inference, 0.3 ms post-processing), or 26.6 frames per second on a laptop NVIDIA RTX 5060, which is comfortably within the budget of a survey-rate aerial pipeline.

The averaged figures conceal the result that matters most here. The two rider classes are the two weakest by AP50-95 (bicycle 0.167, motor 0.326), and bicycle is markedly the weakest of all five: recall 0.308 and AP50 0.329, against an AP50 of 0.909 for small_vehicle, the strongest class. Fewer than one annotated pedal cyclist in three is found. Two factors combine. bicycle is the rarest class in training by a wide margin, with 10,477 instances against 169,814 for small_vehicle, a ratio of sixteen to one. A pedal cyclist seen from survey altitude is also among the smallest objects in the frame. motor, nearly three times as frequent, reaches recall 0.632 and AP50 0.642. This asymmetry matters beyond the detector: riders are the targets of the risk stage and not merely its inputs, so a missed rider is a rider the system never evaluates at all. Section 4.4 returns to this point.

4.3. Risk assessment

Figure 2 walks the pipeline through a single frame: the aerial image, its 42 detections, and the resulting risk field with every detected rider ringed and labelled with its level and index. The frame contains four motorised riders spanning all three bands: one High at I = 0.871, two Medium at 0.463 and 0.326, the latter just above τ_1, and one Low at 0.051. That is a nearly fortyfold spread in accumulated risk S between the most and least exposed rider in the same scene, produced entirely by their positions relative to the surrounding traffic. The High rider's index comes almost entirely from nearby small vehicles, which supply 99.5 % of its accumulated risk; for the four riders together the share lies between 97.1 % and 99.5 %. An index can always be attributed back to named neighbours, which is the practical benefit of the closed-form formulation over a learned end-to-end regressor. The same frame also makes the recall limitation of Section 4.2 concrete: a motorcyclist in the lower centre of the image is not detected, and so receives no index at all.

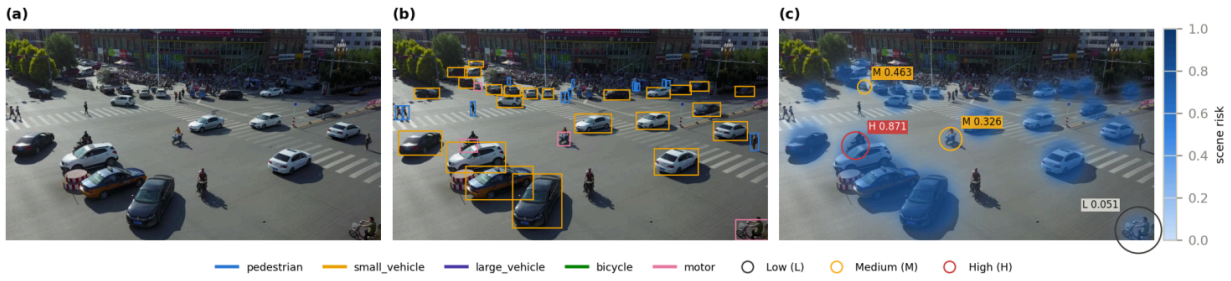


Figure 2. The pipeline applied to one validation frame (0000301_00201_d_0000157). (a) The aerial image. (b) The 42 detections: 14 pedestrians, 24 small vehicles and 4 motorised riders. (c) The display risk field, the maximum of the per-detection risk functions over a grid (Section 2.3.2), with each detected rider marked by a thin hollow ring coloured by level (Low dark grey, Medium amber, High red) and labelled with its level letter and risk index. The four riders score 0.871 (High), 0.463 and 0.326 (Medium) and 0.051 (Low). A motorcyclist in the lower centre of the frame is missed by the detector and is therefore not scored

Applying the pipeline to the whole validation split yields 4,732 riders, 529 pedal cyclists and 4,203 motorised riders, distributed over 487 of the 548 images. The median rider has three contributing neighbours, and 46 riders have no neighbour at all and receive I = 0. The normalisation of Section 2.3.2 gives ρ_0 = 0.111,

and the two thresholds follow as $\tau_1 = 0.323$ and $\tau_2 = 0.500$, the latter by construction. Across all riders the index has mean 0.247, median 0.212 and maximum 0.927.

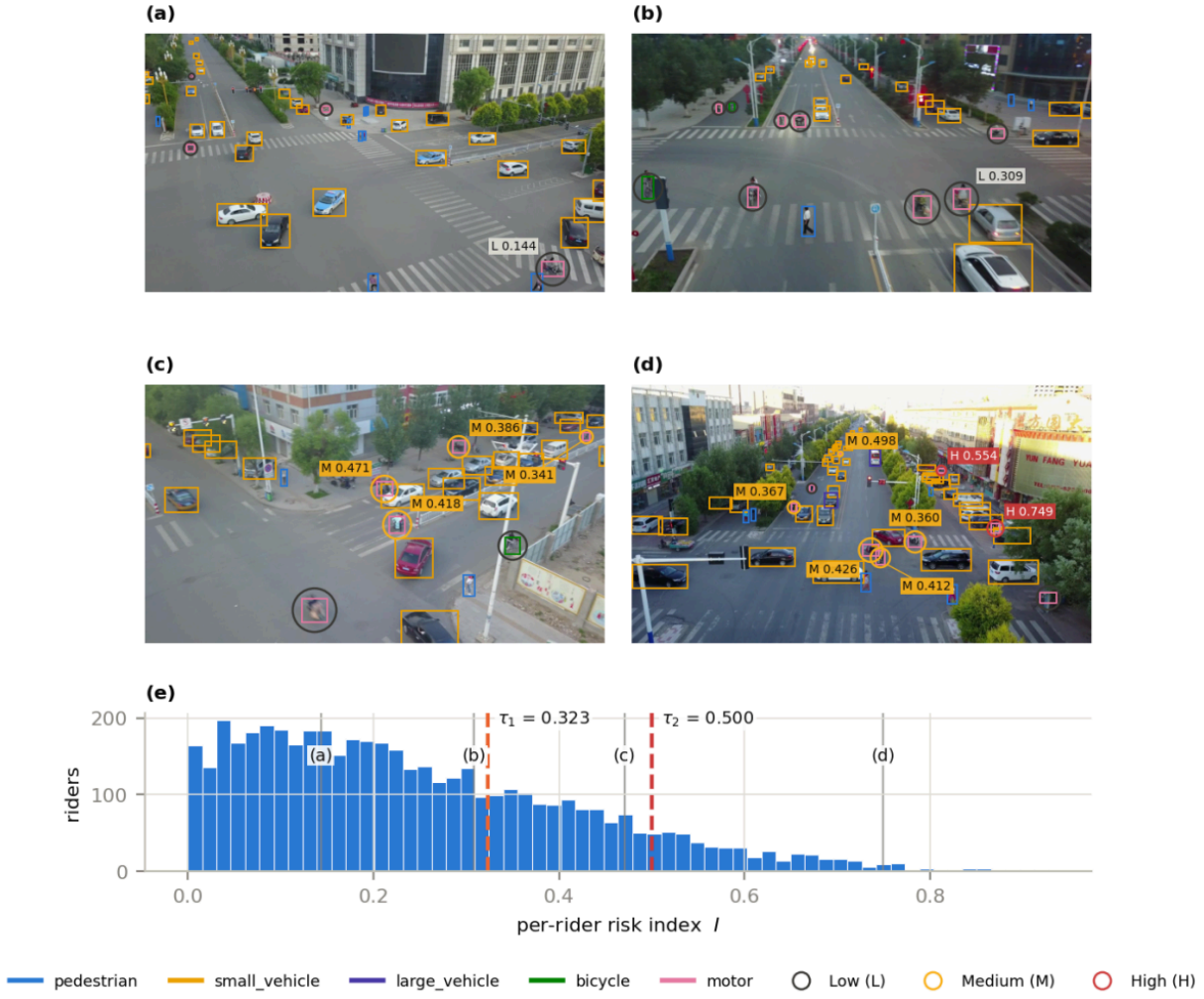


Figure 3. Risk levels across scenes. (a)–(d) Four validation scenes ordered by the index of their most exposed rider; riders are marked by rings coloured as in Figure 2, with detection boxes shown and labels drawn for the riders at or above τ_1 and for the most exposed rider of each panel. (a) 0000330_02001_d_0000810: four riders, all Low, maximum $I = 0.144$. (b) 0000359_00589_d_0000702: nine riders, all Low, maximum $I = 0.309$, just below τ_1. (c) 0000360_02941_d_0000728: six riders (four Medium, two Low), maximum $I = 0.471$. (d) 0000280_03201_d_0000628: nine riders (two High, five Medium, two Low), maximum $I = 0.749$. (e) Distribution of the risk index over all 4,732 detected riders of the validation split; dashed lines mark $\tau_1 = 0.323$ and $\tau_2 = 0.500$, and grey lines the maximum index of scenes (a)–(d)

Figure 3 places four further scenes along this scale before showing where the thresholds come from. In (a) all four riders are Low, the most exposed reaching only $I = 0.144$, well below τ_1. In (b) all nine riders are still Low. The most exposed of them, a rider on a zebra crossing beside a white car, reaches 0.309, just below τ_1. Scene (c) has 30 detections around six riders, four of them Medium, and its maximum of 0.471 lies in the upper part of the Medium band. Scene (d) is the densest, with 59 detections around nine riders. Two of them are High, at 0.749 and 0.554, five are Medium and two are Low, and one Medium rider sits just below τ_2 at

0.498. Panel (e) shows the distribution over all 4,732 riders, with the two thresholds drawn on it and the four scenes marked. The thresholds are read off this distribution at the 70th and 90th percentiles rather than chosen by hand.

The levels are relative. Fixing them at percentiles fixes the prevalence of each band by construction, so "High" means a rider unusually exposed for ordinary VisDrone traffic, and carries no quantified probability of collision. The ordering of riders is robust to the choices made in Section 2.3. Recomputing the index over 120 scenes under eight alternative settings leaves the rank correlation with the default setting at 0.968 or above in every case. The settings are the two other definitions of the characteristic length, decay exponents of 1.5 and 2.5, a bottom-centre anchor, neighbourhood radii of $\kappa = 5$ and $\kappa = 12$, and a halved distance floor. The worst case is the longest-side definition of L, and the other seven variants all reach 0.971 or above. The absolute values of the index depend on parameters set by judgement, while the ranking they induce does not. The two rider types are practically indistinguishable in exposure: median index 0.216 for pedal cyclists against 0.211 for motorised riders, with 9.5 % and 10.1 % respectively in the High band. Pedal cyclists fall slightly more often in the Medium band (23.3 % against 19.6 %). The absence of any systematic difference supports pooling the two classes into one rider set (Section 2.2.2).

4.4. Discussion and limitations

Missed riders are the dominant limitation of the system. At recall 0.308 for pedal cyclists and 0.632 for motorised riders, 2,611 of the 6,173 annotated riders are absent from the risk stage entirely: they are never scored at all. Size accounts for much of this. The riders the detector misses are systematically smaller than those it finds, with a median characteristic length of 17.5 pixels against 22.6. Riders below ten pixels are recovered in fewer than half of cases, against roughly two thirds for larger ones; these are objects a few pixels across in a 1,360-pixel-wide frame. The riders that are found are scored soundly. The neighbours that drive the index are vehicles, and vehicles are the classes the detector recovers most reliably: small_vehicle, which supplies 97.1–99.5 % of a typical rider's index, reaches recall 0.858 and AP50 0.909. For the 3,602 riders matched between detections and labels, the index computed from detections correlates with the index computed from ground truth at $r = 0.849$, and the two assign the same level in 79.0 % of cases. The system is thus accurate about the riders it sees and says nothing about a substantial fraction of those it does not. Improving small-object recall would do more for it than any refinement of the risk model.

The risk parameters rest on judgement, which limits what the index can claim. No risk-labelled aerial dataset exists against which the class weights could be fitted, so they are ordered by the mass and speed differential each class presents to a rider. The index therefore orders riders by exposure and makes no statement about the probability of a collision. The stability result of Section 4.3 supports that narrower claim, though it is no substitute for calibration against observed conflicts, which remains future work. The weights also matter less than their number suggests. The squared decay term lets proximity to vehicles dominate the sum, so cruder weights would yield much the same ranking. This is part of why the ranking is as stable as it is, and it indicates that finer weight design is worth doing only once the inputs improve.

The method is single-frame and therefore blind to motion. It has no access to speed, heading or time-to-collision: a rider moving away from a vehicle scores exactly as one closing on it. This is the clearest motivation for the multi-frame extension proposed in Section 5. No camera geometry is used either, so genuinely oblique frames are treated as if they were near-nadir. The scale-free risk function removes the need for altitude, but VisDrone supplies no intrinsics with which to correct for the viewing angle.

Excluding people, tricycle and awning-tricycle (Section 3.1) makes the computed neighbourhoods slightly sparser than the real traffic. Motorised delivery tricycles in particular are common in Chinese cities and

behave much like two-wheelers as a hazard source. One point of deployment context should be added. In cities that restrict motorcycles from urban districts, the motor detections the system flags are in practice the electric two-wheelers of policy interest. That is a way of reading the output, and not a property of the VisDrone data, whose per-image city labels are not published.

The label set itself is part of the research question. An earlier iteration of this project used a class mapping that kept motorised two-wheelers but had no bicycle class. That mapping would have left the system unable to answer the question it was built to answer. Correcting it and retraining was a precondition for every number reported here.

5. Conclusions

This paper has presented a framework for estimating, from a single UAV frame, how much traffic risk each rider of a two-wheeled vehicle is exposed to. Its contribution is the method itself. A detector supplies the scene content. Every neighbour within a rider-relative neighbourhood contributes a hazard term proportional to the solid angle it subtends at the rider and to how dangerous its class is to a rider. The accumulated hazard is mapped onto a bounded index and, by percentiles of its own empirical distribution, onto a Low / Medium / High level. Every step after detection is closed-form, and its parameters follow from the geometry of the aerial view and from stated conventions. The framework can therefore be re-instantiated with another detector, another class set or another set of weights, and any index it produces can be decomposed back into the specific neighbours that produced it.

Applied to the validation split of VisDrone2019-DET, the framework assigns an index and a level to every detected rider at real-time frame rates on a single laptop GPU. The ordering it induces remains stable under substantial changes to its parameters. This supports the narrow claim the method makes, that it ranks riders by exposure, and it shows that a rule-based formulation is sufficient for that task. The limiting factor is perception. The detector finds fewer than one annotated pedal cyclist in three, so many of the riders the framework is meant to protect are never scored at all. Improving the perception stage therefore buys more than refining the risk model.

Several directions follow from this. First, small-object recall must improve; tiling the high-resolution frames at inference and oversampling the rare classes during training are the obvious first attempts. Second, multi-frame video and object tracking would supply the speed and heading the present formulation lacks, and with them genuine time-to-collision. Third, a dataset that separates electric from fuel-powered two-wheelers would allow the two rider populations to receive distinct parameters, which the present data cannot support. Fourth, aggregating per-rider indices over a survey flight into spatial risk-point maps is the form in which the output would reach a traffic authority, and is a short step from what is already computed.

References

[1] Jia, B., Li, J., & Wang, Q. (2025). Understanding electric bike accidents through safe system approach in Guangzhou, China: A mixed-methods study. *Systems, 13*(4), 261.

[2] Chen, Y., Tao, Z., Chen, Q., He, J., Ruan, X., & Ling, X. (2026). Investigating risky behaviors and safety countermeasures for e-bike riders in China: A traffic conflict analysis approach. *Systems, 14*(1), 37.

[3] Lepin, F., Velastin, S. A., León, R., García-Herrero, J., Rojas-Martínez, G., & Espinosa-Oviedo, J. E. (2026). UTUAV: A drone dataset for urban traffic analysis. *Drones, 10*(1), 15.

[4] Zhu, P., Wen, L., Du, D., Bian, X., Fan, H., Hu, Q., & Ling, H. (2022). Detection and tracking meet drones challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 44*(11), 7380–7399.

[5] Outay, F., Mengash, H. A., & Adnan, M. (2020). Applications of unmanned aerial vehicle (UAV) in road safety, traffic and highway infrastructure management: Recent advances and challenges. *Transportation Research Part A: Policy and Practice, 141*, 116–129.

[6] Zong, S., Chen, S., Alinizzi, M., Li, Y., & Labi, S. (2021). *Using UAVs for vehicle tracking and collision risk assessment at intersections*. arXiv. https: //doi.org/10.48550/arXiv.2110.06775

[7] Jocher, G., Qiu, J., & Chaurasia, A. (2026). *Ultralytics YOLO26* (Version 8.4.33) [Computer software]. Ultralytics. https: //github.com/ultralytics/ultralytics

[8] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779–788.

[9] Wang, J., Wu, J., & Li, Y. (2015). The driving safety field based on driver–vehicle–road interactions. *IEEE Transactions on Intelligent Transportation Systems, 16*(4), 2203–2214.

[10] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. *Proc. European Conference on Computer Vision (ECCV), Lecture Notes in Computer Science 8693*, 740–755.