How Does Perception Affect Safety: New Metrics and Strategy

Xiaotong Zhang, Jinger Chong and Kamal Youcef-Toumi

Abstract— Perception plays a pivotal role in enhancing the functionality of autonomous agents. However, the intricate relationship between robotic perception metrics and actuation metrics remains unclear, leading to ambiguity in the development and fine-tuning of perception algorithms. In this paper, we introduce a methodology for quantifying this relationship, taking into account factors such as detection rate, detection quality, and latency. Furthermore, we introduce two novel perception metrics for Human-Robot Collaboration safety predicated upon basic perception metrics: Critical Collision Probability (CCP) and Average Collision Probability (ACP). To validate the utility of these metrics in facilitating algorithm development and tuning, we develop an attentive processing strategy that focuses exclusively on key input features. This approach significantly reduces computational time while preserving a similar level of accuracy. Experimental findings demonstrate that integrating this strategy into an object detector results in a notable maximum reduction of 30.09% in inference time and 26.53% in total time per frame. Additionally, the strategy lowers the CCP and ACP in a baseline model by 11.25% and 13.50%, respectively.

I. INTRODUCTION

Intelligent agents and machines, such as robots [1], [2], autonomous vehicles, and machines [3], [4], [5], [6], [7], are becoming increasingly important in human daily life and industrial applications. They require perception algorithms to sense and understand the dynamic environment fast and accurately, dynamic planning algorithms to proactively update the plan, and control algorithms to generate the control commands [8]. This is the so-called pipeline frameworks for autonomous agents. Perception is an important but computationally expensive component.

However, an intensive literature review shows that several issues and shortcomings still exist. First, the relationship between the metrics of perception tasks and the metrics of the robot tasks is unclear. Researchers working on a specific perception problem focus on improving the metrics for the specific perception task. However, the ultimate goal of perception algorithms is for real-world applications. Thus, a model, that is beneficial for understanding the effect of the perception performance on robot performance, is desired and necessary. Second, the accuracy-speed-tradeoff is an inevitable issue when developing and tuning perception models. Neither a perfectly accurate algorithm with a slow inference speed nor a fast algorithm with a very low accuracy

All authors are with the Mechatronics Research Laboratory, Massachusetts Institute of Technology, Cambridge, MA, 02139, USA {kevxt, jinger, youcef}@mit.edu

is desired for autonomous agents. However, there are no models for understanding this problem and no guidelines for finding the optimal balance points when tuning models. Third, many papers and benchmarks in perception focused on and reported the accuracy aspect of metrics, such as classification accuracy, precision, recall, F-1 score, AUC-ROC, etc. We would like to argue, in this paper, that the speed aspect is crucial in applications that require real-time decision-making and reaction to the dynamic environment, such as robotics, autonomous vehicles, remote surgery, surveillance, etc.

In order to mitigate or address the above-mentioned issues, we develop and propose a methodology to model the robot performance metrics based on basic perception task metrics. We first separate and decouple the basic perception metrics into three categories, namely the algorithm's detection rate, detection quality, and latency. Then, we investigate and discuss their influence on robot safety metrics separately. Here, we take the safety issues in Human-Robot Collaboration (HRC) tasks as an example. We propose two new perception metrics for HRC, called Critical Collision Probability (CCP) and Average Collision Probability (ACP), which link the basic perception metrics to the safety performance of the robot in real-world deployment and can be used to optimize perception tasks. We preliminarily validate our modeling with the analysis of the attentive processing strategy. The trends of the model agree well with realistic situations.

To demonstrate the usefulness of the modeling and the metrics, we further develop and propose a new strategy to improve the safety of HRC benefiting from our models by enhancing the inference speed of the algorithm while maintaining similar accuracy, which is inspired by how human brains process information. One essential difference between human beings and machines is that human beings weigh the priority and importance of different tasks and inputs, process the essential components with more resources, and dynamically adjust the focus based on task characteristics or other guidance. This mechanism has been studied extensively and verified with examples, e.g., foveal-peripheral vision [9], [10], inattentional blindness [11], change blindness [12], etc.

Inspired by these functions and their effectiveness in human perception, it would be desirable for machines to selectively process the essential components determined by the characteristics and states of the tasks and surrounding agents. Another inspiration of this attentive processing method is the nature of visual inputs, that only a small portion of the visual inputs are related to the task or the short-term decision making. Thus, respecting this nature can dramatically decrease the computation resources required, accelerating the model inference or reducing the hardware requirements and energy

consumption for large model inference.

To this end, we develop and propose a new generic strategy applicable to most computer vision tasks, i.e., attentive processing. In this strategy, the algorithm focuses on and processes only the essential components related to the task with an ensemble of neural networks with different scales. Firstly, we dynamically update the attentive region based on the results from previous frames, respecting the continuity of agent states in real-world applications. Secondly, only the local information within the attentive region is fed into the neural network with the corresponding scale to reduce unnecessary processing in a larger-scale network. Thirdly, the processing of the attentive region results is mapped back to the global input frame. Then, information from the current frame is processed with a much more efficient and lightweight neural network instead of processing every pixel in the global frame with a denser and deeper neural network. The results of the current frame can guide the selection of the attentive region for the next frame. This strategy can be used along with most computer vision algorithms as an additional layer to select only the essential components for processing. In this paper, we use a video object detection task as a case study for verification and validation purposes.

To summarize, the contributions of this work are three-fold: (i) A methodology for modeling robot performance metrics based on perception metrics. Here, we use HRC safety metrics considering object detection recall, IoU, and latency, as a case study. (ii) We model and propose two novel comprehensive perception metrics considering speed-accuracy-tradeoffs, namely ACP and CCP, in the context of HRC tasks, reflecting HRC safety. Thus, optimization of perception algorithms based on our metrics can directly optimize the HRC safety. (iii) A new generic attentive processing strategy on top of efficient learning methods to select, track, and update the essential components in input information for reduced computation resource requirement and effort by utilizing the sparsity in input data. Experiments verify that the attentive processing strategy can enhance safety and speed in human-robot collaboration.

II. Related Work

To the best of our knowledge, our work is very unique. The following sub-sections provide a summary of works closely related to ours.

A. Perception and Robot Metrics

Investigation and analysis of the relationship between perception and robot metrics are crucial. Some previous works analyze the relationship between latency and robot performance [13], [14], [15], [16]. The most related work to ours is [17], which modeled and analyzed the influence of visual latency on high-speed sense and avoidance in a drone application. This work also demonstrated and verified the effectiveness of event cameras in high-speed applications. However, this work doesn't include detection rate and detection quality in the analysis, and it poses strong assumptions (e.g., unchanged longitudinal velocity) in the analysis, which

are difficult to generalize to other applications. Our work is more comprehensive, including detection rate and quality, and suitable for generic collision avoidance problems.

B. Efficient Deep learning

Accelerating and decreasing the computation resource requirements of model inference is an active research area. Methods, including quantization [18], [19], [20], [21], deep compression [22], knowledge distillation [23], sparsification [24], and pruning [25], [26], [27], [28], have shown promising results in the application of edge devices with limited computational resources. These methods tweak the architecture and the precision of the neural networks themselves to remove unnecessary connections and components, reducing the precision of weights and activations, and/or replacing the large network with a more efficient network. However, none of these utilizes the sparsity of the input features. Our methods can work along with these efficient deep learning methods by distilling only important features for processing.

III. Safety Metric Modeling

A. Problem Setup

The problem setup is shown in Figure 1. In this model, we consider a general HRC problem, where the human subjects focus on their own tasks, and the robot facilitates by conducting some other auxiliary tasks. The robot executes the planned path until a human is detected within a safety range by an object detector or human pose estimator. After the detection, the robot either re-plans the path or stops in urgent situations. In this model, for simplicity, we assume that the robot slows down after detecting, and speeds are assumed constant during the frames considered. Our future work will consider the more complex situations with arbitrary motion of human hands and robots.

Without loss of generality, we assume that the camera is fixed in the world coordinates. Thus, all variables are denoted in the camera coordinates. We denote the gripper as object A with a spherical safety margin s_A and the hand as object B with a spherical safety margin s_B. The parameter s_A, or s_B, is the radius of the safety margin. Here, we assume that the safety margins are the same as their sizes. For the human hand, which is detected by perception algorithms, we distinguish the ground truth B and its estimate $\hat{B}$ to clearly analyze the effect of detection quality. The location of robot A can be derived through encoders and kinematics, whose error is smaller. Here, we only distinguish B and $\hat{B}$ for analysis of detection quality and use $\hat{B}$ to represent the location of B.

In the camera coordinates, the position vectors of A and $\hat{B}$ are $\mathbf{R}_A$ and $\mathbf{R}_{\hat{B}}$, respectively, with velocity vectors $\mathbf{U}_A = \dot{\mathbf{R}}_A$ and $\mathbf{U}_{\hat{B}} = \dot{\mathbf{R}}_{\hat{B}}$. The position vector from A to B is $\mathbf{R}_{A\hat{B}} = \mathbf{R}_{\hat{B}} - \mathbf{R}_A$. The distance between A and B can be calculated as $d_{A\hat{B}} = |\mathbf{R}_{A\hat{B}}|$. Collision in this paper is defined when the condition $d_{A\hat{B}} < s_A + s_B$ is true. The velocity of A relative to B can be calculated as $\mathbf{U}_{A\hat{B}} = \mathbf{U}_A - \mathbf{U}_{\hat{B}}$. The illustration of variables to calculate collision condition after being projected to the plane containing $\mathbf{U}_{A\hat{B}}$ and $\mathbf{R}_{A\hat{B}}$ is
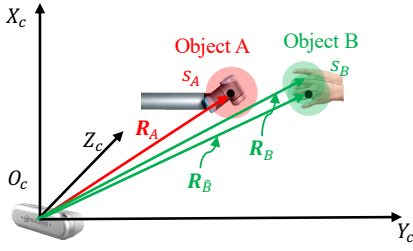
Fig. 1. Problem setup with camera coordinates and necessary variables for analysis.
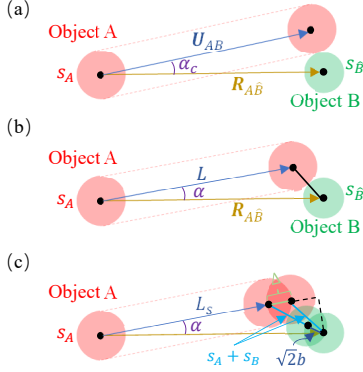

Fig. 2. The illustration of collision condition. (a) critical condition (object A and B collide in a tangent manner); (b) general condition ($\alpha \in [-\alpha_c, \alpha_c]$). (c) general condition after considering the detection shift.

shown in Figure 2. The angle between $\mathbf{R}_{A\hat{B}}$ and $\mathbf{U}_{A\hat{B}}$ is defined as α. The critical condition can be re-written as $\alpha_c = \arctan\left((s_A + s_B)/\sqrt{|\mathbf{R}_{A\hat{B}}|^2 - (s_A + s_B)^2}\right)$. The collision condition can only happen when $\alpha \in [-\alpha_c, \alpha_c]$. Under this condition, the maximum safe travel distance L without considering the bounding box shift can be calculated as follows

$$L = |\mathbf{R}_{A\hat{B}}|\cos\alpha - \sqrt{(s_A + s_B)^2 - |\mathbf{R}_{A\hat{B}}|^2\sin^2\alpha} \quad (1)$$

B. Effect of Latency

The total latency per frame T_t from sensory input to successful response can be decomposed as follows

$$T_t = T_p + T_r \quad (2)$$

where T_p and T_r are the latency for perception and response, respectively. T_r affects the allowable traveling distance before collision. The effect of T_p is reflected in the number of possible frames to detect humans before collision.

C. Effect of Detection Rate

The detection rate considers that the algorithms generate false negatives, which can be fatal. In perception tasks, for one frame, the probability of successful detection equals recall of the algorithm, which means $P_d = Recall$. We assume that frames are independent. Then, we define a random variable k as the average number of frames needed for

the algorithm to make a successful detection. k follows a Geometric Distribution as follows:

$$k \sim Geo(P_d = recall) \quad (3)$$

D. Effect of Detection Quality

Detection quality metrics in perception include Intersection over Union (IoU), Mean Average Precision (mAP), F1 score, etc. Here, we use IoU as the detection quality metric. We assume that the camera is homogeneous in its imaging plane $o'x'y'$ without skewness and distortion. Because of the linear operation in the camera imaging model and the dimensionless property of IoU, IoU for the object on the image plane equals IoU of the object in the world coordinates. Thus, according to the IoU in the image plane, we can derive the distance between the real object and the predicted bounding box in the world coordinate.

The IoU under these assumptions is calculated as follows

$$IoU = \frac{(s_B - b)^2}{2s_B^2 - (s_B - b)^2} \quad (4)$$

where b is the shift between the real location of the object and the predicted location in the world coordinate in the direction of x or y. We can derive the value of b from (4) as

$$b = s_B \left(1 - \frac{\sqrt{2IoU}}{\sqrt{IoU + 1}}\right) \quad (5)$$

Thus, the total maximum shift in the world coordinate is $\sqrt{2}b$. According to the geometrical relationship shown in Figure 2(c), the safe traveling distance decreases by Δ, which is the shift of A along L because of the detection shift of B. Here, we consider the maximum effect of b on L. Thus, we derive the maximum safe traveling distance L_s as

$$L_s = max\left(0, L - \left(\sqrt{(s_A + s_B + \sqrt{2}b)^2 - |\mathbf{R}_{A\hat{B}}|^2\sin^2\alpha}\right.\right.$$
$$\left.\left. - \sqrt{(s_A + s_B)^2 - |\mathbf{R}_{A\hat{B}}|^2\sin^2\alpha}\right) - |\mathbf{U}_{A\hat{B}}|T_r\right) \quad (6)$$

E. Safety Metrics

After considering all the factors discussed above, we could derive the probability of collision P_c as follows:

$$P_c = \begin{cases} P(k > m) = (1 - P_d)^m, & \text{if } \alpha \in [-\alpha_c, \alpha_c] \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

where m is the number of frames processed by the perception algorithm before traveling the distance L_s and calculated as follows:

$$m = \left\lfloor \frac{L_s}{|\mathbf{U}_{AB}| \cdot T_p} \right\rfloor \quad (8)$$

A larger m means more input frames are available to the algorithm before collision.

1) Critical Collision Probability **CCP**: Critical Collision Probability **CCP** is an important metric, which represents the average collision probability under critical and dangerous conditions. CCP is defined as

$$\mathbf{CCP} = \mathbb{E}_{\alpha, |\mathbf{R}_{A\hat{B}}|, |\mathbf{U}_{A\hat{B}}| \in C}[P_c] \tag{9}$$

where $\mathbb{E}$ is the expectation operator, C is the region of critical conditions. In this condition, α, $|\mathbf{R}_{A\hat{B}}|$, and $|\mathbf{U}_{A\hat{B}}|$ may be correlated. An example of the parameter selection is discussed in the Experiments section.

2) Average Collision Probability **ACP**: Further, we define another safety metric Average Collision Probability **ACP**, which is the expectation of collision probability over applicable parameter space representing the average performance over a wide range of conditions, as

$$\mathbf{ACP} = \mathbb{E}_{\alpha, |\mathbf{R}_{A\hat{B}}|, |\mathbf{U}_{A\hat{B}}| \in D}[P_c] \tag{10}$$

where D is the parameter space for the application. The value of ACP represents the average collision probability under all robot operation conditions.

The two metrics can be considered separately or jointly with weight either as an evaluation protocol of perception algorithms or as a training objective for perception algorithms.

IV. ATTENTIVE PROCESSING STRATEGY

The working principle of the proposed attentive processing approach is shown in Figure 3. The main components of the framework can be decomposed into Input Feature Preprocessing (IFP), Efficient Inference (EI), and Result Mapping (RM). We use video object detection, which is important for robotics perception, as an example for illustration purposes.

A. Input Feature Preprocessing

1) Attentive Region Generator: For the current input frame **I**, the attentive regions containing the attentive features are selected by the Attentive Region Generator (AGG) based on past results.

Three methods can serve this purpose: prediction-based, expansion-based, and hybrid. The prediction-based integrates lightweight prediction to predict the attentive region for the current frame based on the bounding box information from the past frames [29]. The expansion-based method sets the attentive regions for the current frame by expanding the bounding box from the last frame so that it has high confidence in including the attentive object in the attentive region. The expansion scale can be dynamic based on the characteristics of bounding boxes in past frames. Hybrid methods can combine both prediction and expansion.

2) Aggregation Method Optimizer: Aggregation Method Optimizer determines the methods to aggregate patches or regions optimally. Two options are stitching regions into one input [30] or processing different regions individually. The second option can also reserve optimality in some cases since we found that twice of the lightweight model inference time may still be faster than the heavyweight model sometimes.

3) Input Feature Aggregation: This module transfers the input frame **I** to **I**′ based on the attentive region selected and the aggregation method selected. After the Input Feature Preprocessing module, the original input **I** is reorganised into a new feature input **I**′.

B. Efficient Inference with Ensemble of Networks

In this component, we scale down the full-resolution object detection network to form an ensemble of models with different input sizes. The set of ensembled neural network input sizes is defined as $\mathbb{K}$. The size of the model selected by the Network Selector (NS) is $o = \min(\mathbb{O})$, where $\mathbb{O}$ is the subset of $\mathbb{K}$ larger than the size of **I**′. Then, **I**′ is processed by the model $M(o)$ with input size o.

C. Result Mapping

In the last step, the results derived from **I**′ are mapped back to the global image **I**. With this step, we obtain the final results in the original input coordinates for other downstream robot tasks.

V. EXPERIMENTS

A. Dataset

We use the LaSOT dataset [31] as the proof-of-concept validation of our strategy on video object detection of one object to verify the effectiveness of our strategy in dramatically reducing the computation complexity while maintaining accuracy and recall. The videos in the LaSOT dataset resemble the RGB input of a general HRC robot working in the wild and with human beings.

To make the dataset applicable to our testing, we screened all video sequences in LaSOT with the following criteria: (1) The object should also be included in the dataset for training the baseline object detector. In this paper, we use You Only Look Once v7 (YOLOv7) [32] as the baseline object detector, which utilizes the COCO dataset [33] while training. For YOLOv7, we use models W6, E6, and D6 as baseline models; (2) LaSOT is a single object tracking dataset, which only contains at most one bounding box on each frame, but perhaps with more than one target object in the frame. To avoid vagueness and confusion on the object detector, we selected the video sequences with only one object from the target class on each frame. (3) To conduct tests on LaSOT without transfer learning, we selected video sequences where at least 50% of objects can be detected with the baseline object detector. (4) The minimum edge length of the image is larger than 640. After the dataset screening, 79 video sequences from LaSOT were selected from 18 categories with a total of 243,076 frames.

B. Implementation Details

The testing is conducted on a TITAN RTX GPU with 24 Gb memory. We use YOLOv7 object detectors with an input size of 1280 by 1280 as the baseline models for all the testing. For all videos, the ensemble of networks is composed of one default model and three other models downscaled from the default model. The confidence threshold in the
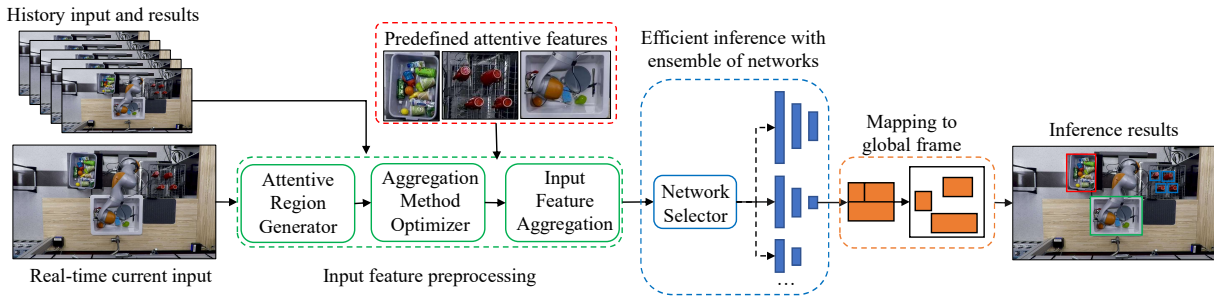
Fig. 3. The proposed attentive processing strategy. The essential and attentive features in the real-time current input $\mathbf{I}$ are selected, cropped, and aggregated into a new feature map $\mathbf{I}'$. Because the dimension of $\mathbf{I}'$ is smaller than $\mathbf{I}$, we use the optimal networks within an ensemble of networks to process $\mathbf{I}'$ efficiently with similar performance accuracy. Finally, the prediction results are mapped back to the global input frame.

model is set as 0.1. The expansion rate for LaSOT in AGG is set to 2.

C. Metrics

For the perception performance evaluation, we use the metrics inference time, total time (inference time + time for other overhead computation), Average Recall (AR), average IoU, and average precision AP at IoU of 0.5 and 0.75. AR and AP are calculated according to COCO evaluation protocol, in percentage.

When computing ACP, we use the following D: $\alpha \in [-\pi, \pi]$, $|\mathbf{R}_{AB}| \in [0.25 \text{ m}, 1.5 \text{ m}]$, and $|\mathbf{U}_{AB}| \in [0.02 \text{ m/s}, 1 \text{ m/s}]$. And $T_r = 0.1$ s. When computing CCP, we use the following C: $|\mathbf{U}_{AB}| \in [0.02 \text{ m/s}, 1 \text{ m/s}]$, $|\mathbf{R}_{AB}| \in [0.25 \text{ m}, \max(0.25m, 0.5s \cdot |\mathbf{U}_{AB}|)]$, and $\alpha \in [-\alpha_c, \alpha_c]$.

D. Results on Perception Metrics

The testing results are summarized in Table I. Our strategy outperforms by a large margin in both inference time and total time per frame, while only sacrificing a fairly small amount of accuracy on all baseline models. The percentage of reduction in inference time and total time increases with the baseline model complexity. After comparing the inference time and total time for each model, it was found out that the overhead computation time for our strategy and baseline models is almost the same. This demonstrates that the components within our framework shown in Figure 3 are very lightweight and fast to compute.

As shown in Figure 4, the average inference time of the neural networks, after scaling down because of the decreased input size, is much shorter. With a high probability, a larger bounding box size ratio (bounding box size over the size of input frame $\mathbf{I}$) corresponds to a larger input size model and, thus, a higher inference time.

During our testing, the main source of accuracy decrease in this preliminary testing is analyzed. It is found that the smaller model, though fast, is the main source of inaccuracy because of the model scaling in this testing. However, our strategy has a strong potential to overcome this bottleneck and achieve improvement in both computation complexity and accuracy by training from scratch or fine-tuning the lower input size models. The lower input size models only require processing the inputs with objects almost in the center
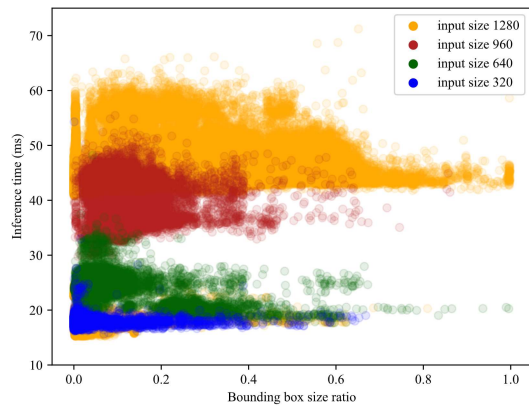

Fig. 4. Inference time with respect to different bounding box size ratios using models with various input sizes. Each dot represents one frame.

and almost with the same scale because of the cropping operation in the framework. This avoids the location variance and scale variance in object detection. Thus, with these methods, the accuracy has the potential to be improved while maintaining efficiency.

E. Results on the Safety Metrics

ACP and CCP with various models are summarized in Table II. With our proposed attentive processing strategy, the ACP and CCP decrease by a large margin, which verifies the effectiveness in improving safety in HRC. Computationally expensive models in the testing are slower to achieve state-of-the-art accuracy and precision on the benchmark. Our strategy works better for those computationally expensive models and reduces CCP and ACP on model D6 by 11.25% and 13.50%, respectively. Even though model D6 achieves the highest IoU, AR, and AP as shown in Table I, its slow processing speed makes model D6 the most dangerous model for HRC as shown in Table II. This validates the importance of examining the speed-accuracy-tradeoffs and developing faster real-time algorithms in robotics applications.

The following analysis is based on model E6. The collision probabilities with various $|\mathbf{U}_{A\hat{B}}|$ and $|\mathbf{R}_{A\hat{B}}|$ are shown in Figure 5 (a) and (b). When the distance $|\mathbf{R}_{A\hat{B}}|$ is short with a high relative velocity $|\mathbf{U}_{A\hat{B}}|$, the collision probability is 1 for both baseline and attentive processing. This agrees
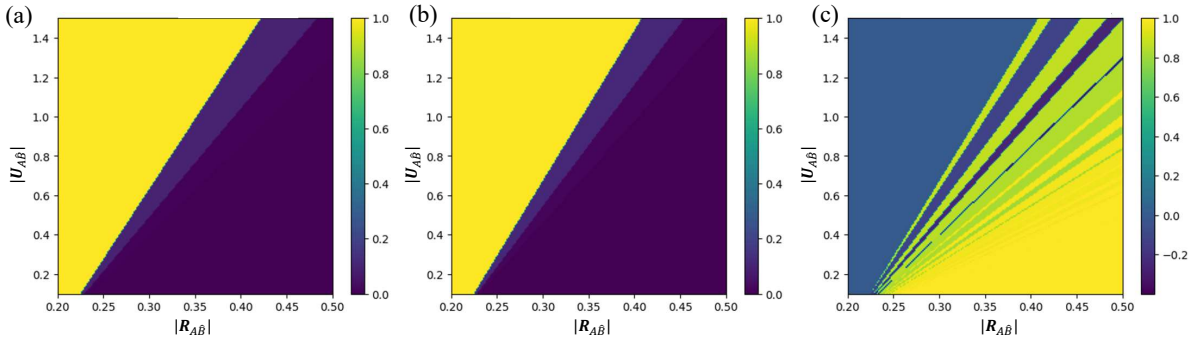
Fig. 5. Collision probability P_c with respect to distance $|\mathbf{R}_{A\hat{B}}|$ and velocity $|\mathbf{U}_{A\hat{B}}|$ with (a) baseline model; (b) attentive processing strategy. The collision probability decrease percentage is in (c). The values are calculated with E6 model perception metrics from Table I. In the plot, we set $\alpha = 0$. The color bars represent collision probability P_c in (a) and (b) and the percentage of P_c decrease in (c).

TABLE I

TESTING RESULTS ON LASOT DATASET WITH THREE BASELINE MODELS. ↑ AND ↓ INDICATE LARGER OR SMALLER IS BETTER. TESTING RESULTS VERIFY THAT OUR STRATEGY IS EFFECTIVE IN REDUCING COMPUTATION TIME WHILE MAINTAINING ACCURACY ON ALL BASELINE MODELS.

Metrics	YOLOv7-W6			YOLOv7-E6			YOLOv7-D6		
	Baseline	Ours	Change %	Baseline	Ours	Change %	Baseline	Ours	Change %
Inference time ↓	24.38	**18.58**	**-23.79%**	34.76	**25.04**	**-27.98%**	42.90	**29.99**	**-30.09%**
total time ↓	30.35	**24.47**	**-19.38%**	40.60	**30.88**	**-23.94%**	48.79	**35.85**	**-26.53%**
overhead time ↓	5.97	5.89	-1.34%	5.83	5.85	0.19%	5.89	5.85	-0.62%
AR ↑	**88.98**	87.97	-1.13%	**89.87**	88.67	-1.33%	**90.38**	89.17	-1.34%
IoU ↑	**0.74**	0.73	-1.22%	**0.75**	0.74	-1.50%	**0.76**	0.74	-1.51%
$AP_{0.5}$ ↑	91.98	**92.02**	**0.03%**	92.58	92.51	**-0.08%**	92.69	92.61	**-0.09%**
$AP_{0.75}$ ↑	**79.39**	78.95	-0.55%	**79.94**	79.49	-0.57%	**80.45**	79.94	-0.63%

TABLE II

COLLISION METRICS WITH VARIOUS MODELS

Model	Metrics	Baseline	Attentive Processing	↓%
W6	CCP	0.46	**0.43**	**6.51%**
	ACP	$5.78 \cdot 10^{-3}$	$\mathbf{5.36 \cdot 10^{-3}}$	**7.27%**
E6	CCP	0.51	**0.46**	**9.20%**
	ACP	$6.54 \cdot 10^{-3}$	$\mathbf{5.83 \cdot 10^{-3}}$	**10.81%**
D6	CCP	0.55	**0.49**	**11.25%**
	ACP	$7.17 \cdot 10^{-3}$	$\mathbf{6.20 \cdot 10^{-3}}$	**13.50%**

well with the realistic case since, in these circumstances, the robot doesn't have sufficient time to process the perception or avoid the obstacle. As the velocity decreases or the distance increases, the collision probability decreases in a step function style because the values of m in equation (8) are integer and discrete. The collision probability is very small for a large distance and small velocity, which also agrees well with the realistic case. The modeling of collision probability is preliminarily validated.

The collision probability decrease in percentage after adopting our strategy is shown in Figure 5 (c). When the collision probabilities are 1, the change percentage is 0. The main difference between the attentive processing and the baseline is in the computation time T_p, recall AR, and IoU. As the distance increases and/or the velocity decreases, the collision probability with attentive processing decreases from 1 first, which causes a jump in the percentage decrease. The reason for this is the short computation time T_p so that the robot can obtain the information for at least one frame with short $|\mathbf{R}_{A\hat{B}}|$ or large $|\mathbf{U}_{A\hat{B}}|$. However, when the baseline can

process the same amount of frames, the collision probability with attentive processing is higher due to the lower accuracy, which is the reason for the negative decrease percentages in some cases. However, these cases become rare when the distance and the velocity further increases and decreases, respectively. Our model of the metric successfully represents this accuracy and speed tradeoff.

VI. CONCLUSION

In this paper, we first develop a methodology to understand and model the relationship between robot metrics and three major categories of perception metrics, i.e. detection rate, detection quality, and latency. We further develop and propose two new HRC perception metrics based on basic perception metrics. To underscore the effectiveness of our methodology, we develop and propose a novel attentive processing strategy that selects the attentive regions containing the essential components in the input, processes the attentive regions efficiently with an optimal model within an ensemble of networks, and finally maps the results back onto the input frame. Our strategy consistently outperforms the baseline models in speed and significantly reduces the total computation time and inference time of an object detector by 26.53% and 30.09% while maintaining a similar level of accuracy. Experiments verify that our strategy dramatically reduces CCP and ACP by 11.25% and 13.50%, respectively, enhancing HRC safety. With this work, the importance of fast and real-time processing is also demonstrated and verified. Hence, we have successfully addressed the three concerns raised in Section I.

REFERENCES

[1] Y. F. Yeung, F. Xia, J. Covarrubias, M. Furokawa, T. Hirano, and K. Youcef-Toumi, "Robotic method and instrument to efficiently synthesize faulty conditions and mass-produce faulty-conditioned data for rotary machines," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 11 382–11 388.

[2] Y. F. Yeung and K. Youcef-Toumi, "An in-pipe manipulator for contamination-less rehabilitation of water distribution pipes," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 6390–6397.

[3] X. Zhang, "Design and optimization of an mhd energy harvester for intelligent pipe systems," Ph.D. dissertation, Massachusetts Institute of Technology, 2019.

[4] X. Zhang and K. Youcef-Toumi, "Magnetohydrodynamic energy harvester for low-power pipe instrumentation," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 6, pp. 4718–4728, 2022.

[5] X. Zhang, A. Al Alsheikh, and K. Youcef-Toumi, "Systematic evaluation and analysis on hybrid strategies of automatic agent last-mile delivery," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 2733–2740.

[6] F. Xia, J. Quigley, X. Zhang, C. Yang, Y. Wang, and K. Youcef-Toumi, "A modular low-cost atomic force microscope for precision mechatronics education," *Mechatronics*, vol. 76, p. 102550, 2021.

[7] J. Qiu, H. Kim, F. Xia, and K. Youcef-Toumi, "Multi-axis active vibration suppression for wafer transfer systems," in *2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2023, pp. 1108–1114.

[8] A. Kothari, T. Tohme, X. Zhang, and K. Youcef-Toumi, "Enhanced human-robot collaboration using constrained probabilistic human-motion prediction," 2023.

[9] J. Xing and D. J. Heeger, "Center-surround interactions in foveal and peripheral vision," *Vision research*, vol. 40, no. 22, pp. 3065–3072, 2000.

[10] C. Vater, B. Wolfe, and R. Rosenholtz, "Peripheral vision in real-world tasks: A systematic review," *Psychonomic bulletin & review*, vol. 29, no. 5, pp. 1531–1557, 2022.

[11] A. Mack, "Inattentional blindness: Looking without seeing," *Current directions in psychological science*, vol. 12, no. 5, pp. 180–184, 2003.

[12] D. J. Simons and D. T. Levin, "Change blindness," *Trends in cognitive sciences*, vol. 1, no. 7, pp. 261–267, 1997.

[13] A. Handa, R. A. Newcombe, A. Angeli, and A. J. Davison, "Real-time camera tracking: When is high frame-rate best?" in *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VII 12*. Springer, 2012, pp. 222–235.

[14] M. Vincze, "Dynamics and system performance of visual servoing," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, vol. 1, 2000, pp. 644–649 vol.1.

[15] S. Behnke, A. Egorova, A. Gloye, R. Rojas, and M. Simon, "Predicting away robot control latency," in *RoboCup 2003: Robot Soccer World Cup VII 7*. Springer, 2004, pp. 712–719.

[16] P. Sermanet, R. Hadsell, J. Ben, A. N. Erkan, B. Flepp, U. Muller, and Y. LeCun, "Speed-range dilemmas for vision-based navigation in unstructured terrain," *IFAC Proceedings Volumes*, vol. 40, no. 15, pp. 300–305, 2007.

[17] D. Falanga, S. Kim, and D. Scaramuzza, "How fast is too fast? the role of perception latency in high-speed sense and avoid," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1884–1891, 2019.

[18] V. Vanhoucke, A. Senior, and M. Z. Mao, "Improving the speed of neural networks on cpus," 2011.

[19] Z. Dong, Z. Yao, A. Gholami, M. W. Mahoney, and K. Keutzer, "Hawq: Hessian aware quantization of neural networks with mixed-precision," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 293–302.

[20] Y. Guo, "A survey on methods and theories of quantized neural networks," *arXiv preprint arXiv:1808.04752*, 2018.

[21] P. Micikevicius, S. Narang, J. Alben, G. Diamos, E. Elsen, D. Garcia, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh, *et al.*, "Mixed precision training," *arXiv preprint arXiv:1710.03740*, 2017.

[22] S. Han, "Efficient methods and hardware for deep learning," Ph.D. dissertation, Stanford University, 2017.

[23] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, pp. 1789–1819, 2021.

[24] X. Zhou, W. Zhang, H. Xu, and T. Zhang, "Effective sparsification of neural networks with global sparsity constraint," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3599–3608.

[25] Y. LeCun, J. Denker, and S. Solla, "Optimal brain damage," *Advances in neural information processing systems*, vol. 2, 1989.

[26] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient inference," *arXiv preprint arXiv:1611.06440*, 2016.

[27] D. Blalock, J. J. Gonzalez Ortiz, J. Frankle, and J. Guttag, "What is the state of neural network pruning?" *Proceedings of machine learning and systems*, vol. 2, pp. 129–146, 2020.

[28] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," *Advances in neural information processing systems*, vol. 28, 2015.

[29] F. Yerlikaya-Özkurt, A. Askan, and G.-W. Weber, "An alternative approach to the ground motion prediction problem by a non-parametric adaptive regression method," *Engineering Optimization*, vol. 46, no. 12, pp. 1651–1668, 2014.

[30] D. Capel, "Image mosaicing," in *Image Mosaicing and super-resolution*. Springer, 2004, pp. 47–79.

[31] H. Fan, L. Lin, F. Yang, P. Chu, G. Deng, S. Yu, H. Bai, Y. Xu, C. Liao, and H. Ling, "Lasot: A high-quality benchmark for large-scale single object tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5374–5383.

[32] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022. [Online]. Available: https://arxiv.org/abs/2207.02696

[33] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context," 2014. [Online]. Available: https://arxiv.org/abs/1405.0312