2

Are Programmers in or out of Control?

The Individual Criminal Responsibility of Programmers of Autonomous Weapons and Self-Driving Cars

MARTA BO[*]

I Introduction

In March 2018, a Volvo XC90 vehicle that was being used to test Uber's emerging automated vehicle technology killed a pedestrian crossing a road in Tempe, Arizona.[1] At the time of the incident, the vehicle was in "autonomous mode" and the vehicle's safety driver, Rafaela Vasquez, was allegedly streaming television onto their mobile device.[2] In November 2019, the National Transportation Safety Board found that many factors contributed to the fatal incident, including failings from both the vehicle's safety driver and the programmer of the autonomous system, Uber.[3] Despite Vasquez later being charged with negligent manslaughter

[*] I am grateful for the helpful discussions within the DILEMA team at the Asser Institute and the feedback received on an earlier draft during the author workshop within the SNSF project "Human–Robot Interaction in Law and its Narratives: Legal Blame, Criminal Law, and Procedure." I thank James Patrick Sexton for his invaluable research assistance and for helping me improve earlier drafts of this chapter. All errors remain mine.

[1] Sam Levin & Julia Carrie Wong, "Self-Driving Uber Kills Arizona Woman in First Fatal Crash Involving Pedestrian," *The Guardian* (March 19, 2018), www.theguardian.com/technology/2018/mar/19/uber-self-driving-car-kills-woman-arizona-tempe ["Self-Driving Uber"]; see also Chapters 6 and 15 in this volume.

[2] Lucia Binding, "Arizona Uber Driver Was 'Streaming The Voice' Moments Before Fatal Crash," *Sky News* (June 22, 2018), https://news.sky.com/story/arizona-uber-driver-was-streaming-the-voice-moments-before-fatal-crash-11413233. In this chapter, I will use interchangeably the terms "driver," "occupant," "operator," and "user."

[3] *Highway Accident Report: Collision between Vehicle Controlled by Developmental Automated Driving System and Pedestrian Tempe, Arizona March 18, 2018* (National Transportation Safety Board, 2019), www.ntsb.gov/investigations/AccidentReports/Reports/HAR1903.pdf.

in relation to the incident,[4] criminal investigations into Uber were discontinued in March 2019.[5] This instance is particularly emblematic of the current tendency to consider responsibility for actions and decisions of autonomous vehicles (AVs) as lying primarily with users of these systems, and not programmers or developers.[6]

In the military realm, similar issues have arisen. For example, it is alleged that in 2020 an autonomous drone system, the *STM Kargu-2*, may have been used during active hostilities in Libya,[7] and that such autonomous weapons (AWs) were programmed to attack targets without requiring data connectivity between the operator and the use of force.[8] Although AW technologies have not yet been widely used by militaries, for several years, governments, civil society, and academics have debated their legal position, highlighting the importance of retaining "meaningful human control" (MHC) in decision-making processes to prevent potential "responsibility gaps."[9] When debating MHC over AWs as well as responsibility issues, users or deployers are more often scrutinized than programmers,[10] the latter being considered too far removed from the effects

[4] *State of Arizona* v. *Rafael Stuart Vasquez*, Indictment 785 GJ 251, Superior Court of the State of Arizona in and for the County of Maricopa (August 27, 2020), www.maricopacountyattorney.org/DocumentCenter/View/1724/Rafael-Vasquez-GJ-Indictment [*State of Arizona*].

[5] "Uber 'Not Criminally Liable' for Self-Driving Death," *BBC News* (March 6, 2019), www.bbc.com/news/technology-47468391.

[6] Manufacturers of AVs often include responsibility clauses in their contracts with end-users; however, practice may vary: see Keri Grieman, "Hard Drive Crash: An Examination of Liability for Self-Driving Vehicles" (2018) 9:3 *Journal of Intellectual Property, Information Technology and E-Commerce Law* 294 ["Hard Drive Crash"] at para. 29.

[7] Letter dated March 8, 2021 from the Panel of Experts on Libya established pursuant to resolution 1973 (2011) addressed to the President of the Security Council (United Nations Security Council, 8 March 2021) S/2021/229, at paras 63–64.

[8] Ibid. at para. 63.

[9] See Filippo Santoni de Sio & Jeroen van den Hoven, "Meaningful Human Control over Autonomous Systems: A Philosophical Account" (2018) 5 *Frontiers in Robotics and AI* 1 ["MHC over Autonomous Systems"] at 10; "Killer Robots and the Concept of Meaningful Human Control: Memorandum to Convention on Conventional Weapons (CCW) Delegates" (Human Rights Watch, 2016), www.hrw.org/news/2016/04/11/killer-robots-and-concept-meaningful-human-control; "Artificial Intelligence and Machine Learning in Armed Conflict: A Human-Centred Approach" (International Committee of the Red Cross, 2019), www.icrc.org/en/document/artificial-intelligence-and-machine-learning-armed-conflict-human-centred-approach.

[10] Berenice Boutin & Taylor Woodcock, "Aspects of Realizing (Meaningful) Human Control: Legal Perspective" in Robin Geiß & Henning Lahmann (eds.), *Research Handbook on Warfare and Artificial Intelligence* (Cheltenham, UK: Edward Elgar, 2024) 9 ["Realizing MHC"] at 2–10.

of AWs. However, programmers' responsibility increasingly features in policy and legal discussions, leaving many interpretative questions open.[11]

To fill this gap in the current debates, this chapter seeks to clarify the role of programmers, understood simply here as a person who writes programmes that give instructions to computers, in crimes committed with and not by AVs and AWs ("AV- and AW-related crimes"). As artificial intelligence (AI) systems cannot provide the elements required by criminal law, i.e. the *mens rea*, the mental element, and the *actus reus*, the conduct element, including its causally connected consequence,[12] the criminal responsibility of programmers will be considered in terms of direct responsibility for commission of crimes, i.e., as perpetrators or co-perpetrators,[13] rather than vicarious or joint responsibility for crimes committed by AI. Programmers could, e.g., be held responsible on the basis of participatory modes of responsibility, such as aiding or assisting users in perpetrating a crime. Despite their potential relevance, participatory modes of responsibility under national and international criminal law (ICL) are not analyzed in this chapter, as that would require a separate analysis of their *actus reus* and *mens rea* standards. Finally, it must be acknowledged that as used in this chapter, the term "programmer" is a simplification. The development of AVs and AWs entails the involvement of numerous actors, internal and external to tech companies, such as developers, programmers, data labelers, component manufacturers, software developers, and manufacturers. These distinctions might entail difficulties in individualizing responsibility and/or a distribution of

[11] Marta Bo, Laura Bruun, & Vincent Boulanin, *Retaining Human Responsibility in the Development and Use of Autonomous Weapon Systems: On Accountability for Violations of International Humanitarian Law Involving AWS* (Stockholm, Sweden: Stockholm International Peace Research Institute, 2022) at 38 and 39.

[12] See Thomas C. King, Nikkita Aggarwal, Mariarosaria Taddeo *et al.*, "Artificial Intelligence Crime: An Interdisciplinary Analysis of Foreseeable Threats and Solutions" (2020) 26:2 *Science and Engineering Ethics* 89 at 95; see *contra* the work of Gabriel Hallevy, "The Criminal Liability of Artificial Intelligence Entities: From Science Fiction to Legal Social Control" (2010) 4:2 *Akron Intellectual Property Journal* 171; see Chapter 4 in this volume.

[13] Direct commission or principal responsibility under international criminal law also includes joint commission and co-perpetration: Gerhard Werle & Florian Jessberger, *Principles of International Criminal Law* (New York, NY: Oxford University Press, 2020) at paras. 623–659. Co-perpetration as a form of principal responsibility in German criminal law is founded on the concept of "control over whether and how the offense is carried out": Thomas Weigend, "Germany" in Kevin Jon Heller & Markus D. Dubber (eds.), *The Handbook of Comparative Criminal Law* (Redwood City, CA: Stanford University Press, 2011) 252 ["Germany"] at 265 and 266. There is no similar "co-perpetration" mode of liability in the United States.

criminal responsibility, which could be captured by participatory modes of responsibility.[14]

This chapter will examine the criminal responsibility of programmers through two examples, AVs and AWs. While there are some fundamental differences between AVs and AWs, there are also striking similarities. Regarding differences, AVs are a means of transport, implying the presence of people onboard, which will not necessarily be a feature of AWs. As for similarities, both AVs and AWs depend on object recognition technology.[15] Central to this chapter is the point that both AVs and AWs can be the source of incidents resulting in harm to individuals; AWs are intended to kill, are inherently dangerous, and can miss their intended target, and while AVs are not designed to kill, they can cause death by accident. Both may unintentionally result in unlawful harmful incidents.

The legal focus regarding the use of AVs is on crimes against persons under national criminal law, e.g., manslaughter and negligent homicide, and regarding the use of AWs, on crimes against persons under ICL, i.e., war crimes against civilians, such as those found in the Rome Statute of the International Criminal Court ("Rome Statute")[16] and in the First Additional Protocol to the Geneva Conventions (AP I).[17] A core issue is whether programmers could fulfil the *actus reus*, including the requirement of causation, of these crimes. Given the temporal and spatial gap between programmer conduct and the injury, as well as other possibly intervening causes, a core challenge in ascribing criminal responsibility lies in determining a causal link between programmers' conduct and AV- and AW-related crimes. To determine causation, it is necessary to delve into the technical aspects of AVs and AWs, and consider when and which of their associated risks can or cannot be, in principle, imputable to a programmer.[18] Adopting a preliminary categorization of AV- and AW-related risks based on programmers' alleged control or lack of it over the behavior

[14] See Chapter 4 in this volume.

[15] See Sections II and III.

[16] United Nations, Rome Statute of the International Criminal Court, 2187 UNTS 3 (adopted July 17, 1998, entered into force July 1, 2002) (Rome, Italy: United Nations, 1998) [Rome Statute].

[17] United Nations, Protocol Additional to the Geneva Conventions of 12 August 1949 and Relating to the Protection of Victims of International Armed Conflicts, 1125 UNTS 3 (signed June 8, 1977, entered into force December 7, 1978) (Geneva, Switzerland: United Nations, 1977) [AP I].

[18] Some theories of causation recognize that causation in law is a matter of imputation, i.e., a matter of imputing a result to a criminal conduct: Paul K. Ryu, "Causation in Criminal Law" (1958) 106:6 *University of Pennsylvania Law Review* 773 ["Causation in Criminal Law"] at 785, 795, and 796.

and/or effects of AVs and AWs, Sections II and III consider the different risks and incidents entailed by the use of AVs and AWs. Section IV turns to the elements of AV- and AW-related crimes, focusing on causation tests and touching on *mens rea*. Drawing from this analysis, Section V turns to a notion of MHC over AVs and AWs that incorporates requirements for the ascription of criminal responsibility and, in particular, causation criteria to determine under which conditions programmers exercise causal control over the unlawful behavior and/or effects of AVs and AWs.

II Risks Posed by AVs and Programmer Control

Without seeking to identify all possible causes of AV-related incidents, Section II begins by identifying several risks associated with AVs: algorithms, data, users, vehicular communication technology, hacking, and the behavior of bystanders. Some of these risks are also applicable to AWs.[19]

In order to demarcate a programmer's criminal responsibility, it is crucial to determine whether they ultimately had control over relevant behavior and effects, e.g., navigation and possible consequences of AVs. Thus, the following sections make a preliminary classification of risks on the basis of the programmers' alleged control over them. While a notion of MHC encompassing the requirement of causality in criminal law will be developed in Section V, it is important to anticipate that a fundamental threshold for establishing the required causal nexus between conduct and harm is whether a programmer could understand and foresee a certain risk, and whether the risk that materialized was within the scope of the programmer's "functional obligations."[20]

II.A Are Programmers in Control of Algorithm and Data-Related Risks in AVs?

Before turning to the risks and failures that might lie in algorithm design and thus potentially under programmer control, this section describes the tasks required when producing an AV, and then reviews some of the rules that need to be coded to achieve this end.

The main task of AVs is navigation, which can be understood as the AV's behavior as well as the algorithm's effect. Navigation on roads is mostly

[19] In the context of AVs, the responsibility of manufacturers and programmers might overlap; see "Hard Drive Crash", note 6 above, at para. 29.

[20] See Sections IV and V.

premised on rules-based behavior requiring knowledge of traffic rules and the ability to interpret and react to uncertainty. In AVs, automated tasks include the identification and classification of objects usually encountered while driving, such as vehicles, traffic signs, traffic lights, and road lining.[21] Furthermore, "situational awareness and interpretation"[22] is also being automated. AVs should be able "to distinguish between ordinary pedestrians (merely to be avoided) and police officers giving direction," and conform to social habits and rules by, e.g., "interpret[ing] gestures by or eye contact with human traffic participants."[23] Finally, there is an element of prediction: AVs should have the capability to anticipate the behavior of human traffic participants.[24]

In AV design, the question of whether traffic rules can be accurately embedded in algorithms, and if so who is responsible for translating these rules into algorithms, becomes relevant in determining the accuracy of the algorithm design as well as attributing potential criminal responsibility. For example, are only programmers involved, or are lawyers and/or manufactures also involved? While some traffic rules are relatively precise and consist of specific obligations, e.g., a speed limit represents an obligation not to exceed that speed,[25] there are also several open-textured and context-dependent traffic norms, e.g., regulations requiring drivers to drive carefully.[26]

AV incidents might stem from a failure of the AI to identify objects or correctly classify them. For example, the first widely reported incident involving an AV in May 2016 was allegedly caused by the vehicle sensor system's failure to distinguish a large white truck crossing the road from the bright spring sky.[27] Incidents may also arise due to failures to correctly

[21] Henry Prakken, "On the Problem of Making Autonomous Vehicles Conform to Traffic Law" (2017) 25:3 *Artificial Intelligence and Law* 341 ["Making Autonomous Vehicles"] at 353.

[22] Ibid.

[23] Ibid. at 354.

[24] Ibid.

[25] See Prakken's analysis of Dutch traffic laws which could be extended to other similar European systems by analogy: "Making Autonomous Vehicles", note 21 above, at 345, 346, and 360. However, Prakken also provides an overview of open-textured and vague norms in Dutch traffic law: ibid. at 347 and 348.

[26] "Making Autonomous Vehicles", note 21 above, at 347 and 348. See the open-textured traffic rules in the *Straßenverkehrsgesetz* (Swiss Traffic Code) (StVG), SR 741.01 (as of January 1, 2020), Arts. 4, 26, and 31, www.admin.ch/opc/de/classified-compilation/19580266/index.html.

[27] Danny Yadron & Dan Tynan, "Tesla Driver Dies in First Fatal Crash While Using Autopilot Mode," *The Guardian* (July 1, 2016), www.theguardian.com/technology/2016/jun/30/tesla-autopilot-death-self-driving-car-elon-musk.

interpret or predict the behavior of others or traffic conditions, which may sometimes be interlinked with or compounded by problems of detection and sensing.[28] In turn, mistakes in both object identification and prediction might occur as a result of faulty algorithm design and/or derived from flawed data. In the former case, *prima vista*, if mistakes in object identification and/or prediction occur due to an inadequate algorithm design, the criminal responsibility of programmers could be engaged.

In relation to the latter, the increasing and almost dominant use of machine learning (ML) algorithms in AVs[29] means that issues of algorithms and data are interrelated. The performance of algorithms has become heavily dependent on the quality of data. A multitude of different algorithms are used in AVs for different purposes, with supervised and unsupervised learning-based algorithms often complementing one another. Supervised learning, in which an algorithm is fed instructions on how to interpret the input data, relies on a fully labeled dataset. Within AVs, the supervised learning models are usually: (1) "classification" or "pattern recognition algorithms," which process a given set of data into classes and help to recognize categories of objects in real time, such as street signs; and (2) "regression," which is usually employed for predicting events.[30] In cases of supervised learning, mistakes can arise from incorrect data annotation instead of a faulty algorithm design per se. If incidents do occur,[31] the programmer arguably would not be able to foresee those risks and be considered in control of the subsequent navigation decisions.

Other issues may arise with unsupervised learning[32] where an ML algorithm receives unlabeled data and programmers "describe the desired behaviour and teach the system to perform well and generalise to new

[28] See e.g., the accident involving a Tesla Model 3 which hit a Ford Explorer pickup truck, killing one passenger: Neal E. Boudette, "Tesla Says Autopilot Makes Its Cars Safer. Crash Victims Say It Kills," *The New York Times* (July 5, 2021), www.nytimes.com/2021/07/05/business/tesla-autopilot-lawsuits-safety.html.

[29] "How Machine Learning Algorithms Made Self Driving Cars Possible?" *upGrad Blog* (November 18, 2019), www.upgrad.com/blog/how-machine-learning-algorithms-made-self-driving-cars-possible/.

[30] See Mindy Support, "How Machine Learning in Automotive Makes Self-Driving Cars a Reality," *Mindy News Blog* (February 12, 2020), https://mindy-support.com/news-post/how-machine-learning-in-automotive-makes-self-driving-cars-a-reality/.

[31] See ibid.

[32] See "What Does Unsupervised Learning Have in Store for Self-Driving Cars?" *intellias* (August 22, 2019), intellias.com/what-does-unsupervised-learning-have-in-store-for-self-driving-cars/.

environments through learning."[33] Data can be provided in the phase of simulating and testing, but also during the use itself by the end-user. Within such methods, "deep learning" is increasingly used to improve navigation in AVs. Deep learning is a form of unsupervised learning that "automatically extracts features and patterns from raw data [such as real-time data] and predicts or acts based on some reward function."[34] When an incident occurs due to deep learning techniques using real data, it must be assessed whether the programmer could have foreseen that specific risk and the resulting harm, or whether it derived, e.g., from an unforeseeable interaction with the environment.

II.B Programmer or User: Who Is in Control of AVs?

As shown in the March 2018 Uber incident,[35] incidents can also derive from failures of the user to regain control of the AV, with some AV manufacturers attempting to shift the responsibility for ultimately failing to avoid collisions onto the AVs' occupants.[36] However, there are serious concerns as to whether an AV's user, who depending on the level of automation is essentially in an oversight role, is cognitively in the position to regain control of the vehicle. This problem is also known as automation bias,[37] a cognitive phenomenon in human–machine interaction, in which complacency, decrease of attention, and overreliance on the technology might impair the human ability to oversee, intervene, and override the system if needed.

Faulty human–machine interface (HMI) design, i.e., the technology which connects an autonomous system to the human, such as a dashboard or interface, could cause the inaction of the driver in the first place. In these instances, the driver could be relieved from criminal responsibility. Arguably, HMIs do not belong to programmers' functional obligations and therefore fall outside of a programmer's control.

[33] Sampo Kuutti, Richard Bowden, Yaochu Jin *et al.*, "A Survey of Deep Learning Applications to Autonomous Vehicle Control" (2021) 22:2 *Institute of Electrical and Electronics Engineers Transactions on Intelligent Transportation Systems* 712 at 713.

[34] Abhishek Gupta, Alagan Anpalagan, Ling Guan *et al.*, "Deep Learning for Object Detection and Scene Perception in Self-Driving Cars: Survey, Challenges, and Open Issues" (2021) 10:10 *Array* 1 at 8.

[35] See "Self-Driving Uber", note 1 above.

[36] See "Hard Drive Crash", note 6 above.

[37] Kathleen L. Mosier & Linda J. Skitka, "Human Decision Makers and Automated Decision Aids: Made for Each Other?" in Raja Parasuraman & Mustapha Mouloua (eds.), *Automation and Human Performance: Theory and Applications* (Boca Raton, FL: CRC Press, 1996) 201 at 203–210.

There are phases other than actual driving where a user could gain control of an AV's decisions. Introducing ethics settings into the design of AVs may ensure control over a range of morally significant outcomes, including trolley-problem-like decisions.[38] Such settings may be mandatorily introduced by manufacturers with no possibility for users to intervene and/or customize them, or they may be customizable by users.[39] Customizable ethics settings allow users "to manage different forms of failure by making autonomous vehicles follow [their] decisions" and their intention.[40]

II.C Are Some AV-Related Risks Out of Programmer Control?

There are a group of risks and failures that could be considered outside of programmer control. These include communications failures, hacking of the AV by outside parties, and unforeseeable bystander behavior. One of the next steps predicted in the field of vehicle automation is the development of software enabling AVs to communicate with one another and to share real-time data gathered from their sensors and computer systems.[41] This means that a single AV "will no longer make decisions based on information from just its own sensors and cameras, but it will also have information from other cars."[42] Failures in vehicular communication technologies[43] or inaccurate data collected by other AVs cannot be attributed to a single programmer, as they might fall beyond their responsibilities and functions, and also beyond their control.

Hacking could also cause AV incidents. For example, "placing stickers on traffic signs and street surfaces can cause self-driving cars to ignore speed restrictions and swerve headlong into oncoming traffic."[44] Here,

[38] See Sadjad Soltanzadeh, Jai Galliott, & Natalia Jevglevskaja, "Customizable Ethics Settings for Building Resilience and Narrowing the Responsibility Gap: Case Studies in the Socio-Ethical Engineering of Autonomous Systems" (2020) 26:5 *Science and Engineering Ethics* 2693 ["Customizable Ethics"] at 2696.

[39] Ibid. at 2705.

[40] Ibid. at 2697.

[41] Kim Harel, "Self-Driving Cars Must Be Able to Communicate with Each Other," *Aarhus University Department of Electrical and Computer Engineering: News* (June 2, 2021), https://ece.au.dk/en/currently/news/show/artikel/self-driving-cars-must-be-able-to-communicate-with-each-other/.

[42] Ibid.

[43] See, on this topic, M. Nadeem Ahangar, Qasim Z. Ahmed, Fahd A. Kahn *et al.*, "A Survey of Autonomous Vehicles: Enabling Communication Technologies and Challenges" (2021) 21:3 *Sensors* 706.

[44] Keith J. Hayward & Matthijs M. Maas, "Artificial Intelligence and Crime: A Primer for Criminologists" (2021) 17:2 *Crime Media Culture* 209 at 216.

the criminal responsibility of a programmer could depend on whether the attack could have been foreseen and whether the programmer should have created safeguards against it. However, the complexity of AI systems could make them more difficult to defend from attacks and more vulnerable to interference.[45]

Finally, imagine an AV that correctly follows traffic rules, but hits a pedestrian who unforeseeably slipped and fell onto the road. Such unforeseeable behavior of a bystander is relevant in criminal law cases on vehicular homicide, as it will break the causal nexus between the programmer and the harmful outcome.[46] In the present case, it must be determined which unusual behavior should be foreseen at the stage of programming, and whether standards of foreseeability in AVs should be higher for human victims.

III Risks Posed by AWs and Programmer Control

While not providing a comprehensive overview of the risks inherent in AWs, Section III follows the structure of Section II by addressing some risks, including algorithms, data, users, communication technology, hacking and interference, and the unforeseeable behavior of individuals in war, and by distinguishing risks based on their causes and programmers' level of control over them. While some risks cannot be predicted, the "development of the weapon, the testing and legal review of that weapon, and th[e] system's previous track record"[47] could provide information about the risks involved in the deployment of AWs. Some risks could be understood and foreseen by the programmer and therefore be considered under their control.

III.A Are Programmers in Control of Algorithm and Data-Related Risks in AWs?

Autonomous drones provide an example of one of the most likely applications of autonomy within the military domain,[48] and this example will be

[45] Matthew Caldwell, Jerone T. A. Andrews, Thomas Tanay *et al.*, "AI-Enabled Future Crime" (2020) 9:1 *Crime Science* 14 at 22.

[46] See Section IV.

[47] Arthur Holland Michel, *Known Unknowns: Data Issues and Military Autonomous Systems* (Geneva, Switzerland: UN Institute for Disarmament Research, 2021) [*Known Unknowns*] at 10.

[48] Merel Ekelhof & Giacomo Persi Paoli, *Swarm Robotics: Technical and Operational Overview of the Next Generation of Autonomous Systems* (Geneva, Switzerland: United Nations Institute for Disarmament Research, 2020) at 51.

used to highlight the increasingly autonomous tasks in AWs. This section will address the rules to be programmed, and identify where some risks might lie in the phase of algorithm design.

The two main tasks being automated in autonomous drones are: (1) navigation, which is less problematic than on roads and a relatively straightforward rule-based behavior, i.e., they must simply avoid obstacles while in flight; and (2) weapon release, which is much more complex as "ambiguity and uncertainty are high when it comes to the use of force and weapon release, bringing this task in the realm of expertise-based behaviours."[49] Within the latter, target identification is the most important function because it is crucial to ensure compliance with the international humanitarian law (IHL) principle of distinction, the violation of which could also cause individual criminal responsibility for war crimes. The principle of distinction establishes that belligerents and those executing attacks must distinguish at all times between civilians and combatants, and not target civilians.[50] In target identification, the two main automated tasks are: (1) object identification and classification on the basis of pattern recognition;[51] and (2) prediction, e.g., predicting that someone is surrendering, or based on the analysis of patterns of behavior, predicting that someone is a lawful target.[52]

Some of the problems in the algorithm design phase may derive from translating the open-textured and context-dependent[53] rules of IHL,[54] such as the principle of distinction, into algorithms, and from incorporating programmer knowledge and expert-based rules,[55] such as those needed to analyze patterns of behavior in targeted strikes and translate them into code.

There are some differences compared with the algorithm design phase in AVs. Due to the relatively niche and context-specific nature of IHL,

[49] Andree-Anne Melancon, "What's Wrong with Drones? Automatization and Target Selection" (2020) 31:4 *Small Wars and Insurgencies* 801 ["What's Wrong"] at 806.

[50] The principle of distinction is enshrined in AP I, note 17 above, at Art. 48, with accompanying rules at Arts. 51 and 52.

[51] Ashley Deeks, "Coding the Law of Armed Conflict: First Steps" in Matthew C. Waxman & Thomas W. Oakley (eds.), *The Future Law of Armed Conflict* (New York, NY: Oxford University Press, 2022) 41 ["First Steps"]; "What's Wrong", note 49 above, at 12 and 13.

[52] E.g., autonomous drones equipped with autonomous or automatic target recognition (ATR) software to be employed for targeted killings of alleged terrorists.

[53] "First Steps", note 51 above, at 53.

[54] On the challenges, see Alan L. Schuller, "Artificial Intelligence Effecting Human Decisions to Kill: The Challenge of Linking Numerically Quantifiable Goals to IHL Compliance" (2019) 15:1–2 *Journal of Law and Policy for the Information Society* 105.

[55] "What's Wrong", note 49 above, at 14–16.

compared to traffic law which is more widely understood by programmers, programming IHL might require a stronger collaboration with outside expertise, i.e., military lawyers and operators.

However, similar observations to AVs can be made in relation to supervised and unsupervised learning algorithms. *Prima vista*, if harm results from mistakes in object identification and prediction based on an inadequate algorithm design, the criminal responsibility of the programmer(s) could be engaged. Depending on the foreseeability of such data failures to the programmer and the involvement of third parties in data labeling, and assuming mistakes could not be foreseen, criminal responsibility might not be attributable to programmers. Also similar to AVs, the increasing use of deep learning methods in AWs makes the performance of algorithms dependent on both the availability and accuracy of data. Low quality and incorrect data, missing data, and/or discrepancies between real and training data may be conducive to the misidentification of targets.[56] When unsupervised learning is used in algorithm design, environmental conditions and armed conflict-related conditions, e.g., smoke, camouflage, and concealment, may inhibit the collection of accurate data.[57] As with AVs, programmers of AWs may at some point gain sufficient knowledge and experience regarding the robustness of data and unsupervised machine learning that would subject them to due diligence obligations, but the chapter assumes that programmers have not reached that stage yet. In the case of supervised learning, errors in data may lie in a human-generated data feed,[58] and incorrect data labeling could lead to mistakes and incidents that might be attributable to someone, but not to programmers.

III.B Programmer or User: Who Is in Control of AWs?

The relationship between programmers and users of AWs presents different challenges than AVs. In light of current trends in AW development, arguably toward human–machine interaction rather than full autonomy of the weapons system, the debate has focused on the degree of control that militaries must retain over the weapon release functions of AWs.

[56] See *Known Unknowns*, note 47 above, at 4; Joshua Hughes, "The Law of Armed Conflict Issues Created by Programming Automatic Target Recognition Systems Using Deep Learning Methods" (2018) 21 *Yearbook of International Humanitarian Law* 99 at 106 and 107.

[57] *Known Unknowns*, note 47 above, at 6.

[58] *Known Unknowns*, note 47 above, at 4.

However, control can be shared and distributed among programmers and users in different phases, from the design phase to deployment. As noted above, AI engineering in the military domain might require a strong collaboration between programmers and military lawyers in order to accurately code IHL rules in algorithms.[59] Those arguing for the albeit debated introduction of ethics settings in AWs maintain that ethics settings would "enable humans to exert more control over the outcomes of weapon use [and] make the distribution of responsibilities [between manufacturers and users] more transparent."[60]

Finally, given their complexity, programmers of AWs might be more involved than programmers of AVs in the use of AWs and in the targeting process, e.g., being required to update the system or implement some modifications to the weapon target parameters before or during the operation.[61] In these situations, it must be evaluated to what extent a programmer could foresee a certain risk entailed in the deployment and use of an AW in relation to a specific attack rather than just its use in the abstract.

III.C Are Some AW-Related Risks Out of Programmer Control?

In the context of armed conflict, it is highly likely that AWs will be subject to interference and attacks by enemy forces. A UN Institute for Disarmament Research (UNIDIR) report lists several pertinent examples: (1) signal jamming could "block systems from receiving certain data inputs (especially navigation data)"; (2) hacking, such as "spoofing" attacks, might "replace an autonomous system's real incoming data feed with a fake feed containing incorrect or false data"; (3) "input" attacks could "change a sensed object or data source in such a way as to generate a failure," e.g., enemy forces "may seek to confound an autonomous system by disguising a target"; and (4) "adversarial examples" or "evasion," which are attacks that "involve adding subtle artefacts to an input datum that result in catastrophic interpretation error by the machine."[62] In such situations, the issue of criminal responsibility for programmers will depend on the modalities of the adversarial interference, whether it could have been foreseen, and whether the AW could have been protected from foreseeable types of attacks.

[59] "First Steps", note 51 above, at 53 and 54.

[60] "Customizable Ethics", note 38 above, at 2704 and 2705.

[61] Military targeting must be intended as encompassing more than critical functions of weapon release.

[62] *Known Unknowns*, note 47 above, at 7.

Similar to the AV context, failures of communication technology, caused by signal jamming or by failures of communication systems between a human operator and the AI system or among connected AI systems, may lead to incidents that could not be imputed to a programmer.

Finally, conflict environments are likely to drift constantly as "[g]roups engage in unpredictable behaviour to deceive or surprise the adversary and continually adjust (and sometimes radically overhaul) their tactics and strategies to gain an edge."[63] The continuously changing and unforeseeable behavior of opposing belligerents and the tactics of enemy forces can lead to "data drift," whereby changes that are difficult to foresee can lead to a weapon system's failure without it being imputable to a programmer.[64]

IV AV-Related Crimes on the Road and AW-Related War Crimes on the Battlefield

The following section will distil the legal ingredients of crimes against persons resulting from failures in the use of AVs and AWs. The key question is whether the *actus reus*, i.e., the prohibited conduct, including its resulting harm, could ever be performed by programmers of AVs and AWs. The analysis suggests that save for war crimes under the Rome Statute, which prohibit a conduct, the crimes under examination on the road and the battlefield are currently formulated as result crimes, in that they require the causation of harm such as death or injuries. In relation to crimes of conduct, the central question is whether programmers controlled the behavior of an AV or an AW, e.g., the AW's launching of an indiscriminate attack against civilians. In relation to crimes of result, the central question is whether programmers exercise causal control over a chain of events leading to a prohibited result, e.g., death, that must occur in addition to the prohibited conduct. Do programmers exercise causal control over the behavior and the effects of AVs and AWs? Establishing causation of crimes of conduct presents differences compared with crimes of result in light of the causal gap that characterizes the latter.[65] However, this difference is irrelevant in the context of crimes committed with the intermediation

[63] *Known Unknowns*, note 47 above, at 9.

[64] Ibid.

[65] Crimes of conduct "rest on an immediate connection between the harmful action and the relevant harm"; crimes of result "are characterized by a [special and temporal] causal gap between action and consequence": George P. Fletcher, *Basic Concepts of Criminal Law* (New York, NY: Oxford University Press, 1998) [*Basic Concepts*] at 61.

of AI since, be they of conduct or result, they always present a causal gap between a programmer's conduct and the unlawful behavior or effect of an AV and AW. Thus, the issue is whether a causal nexus exists between a programmer's conduct and either the behavior (in the case of crimes of conduct) or the effects (in the case of crimes of result) of AVs and AWs. Sections IV.A and IV.B will describe the *actus reus* of AV- and AW-related crimes, while Section IV.C will turn to the question of causation. While the central question of this chapter concerns the *actus reus*, at the end of this section, I will also make some remarks on *mens rea* and the relevance of risk-taking and negligence in this debate.

IV.A Actus Reus *in AV-Related Crimes*

This section focuses on the domestic criminal offenses of negligent homicide and manslaughter in order to assess whether the *actus reus* of AV-related crimes could be performed by a programmer. It does not address traffic and road violations generally,[66] nor the specific offense of vehicular homicide.[67]

Given the increasing use of AVs and pending AV-related criminal cases in the United States,[68] it seems appropriate to take the Model Penal Code (MPC) as an example of common law legislation.[69] According to the MPC, the *actus reus* of manslaughter consists of "killing for which the person is reckless about *causing* death."[70] Negligent homicide concerns instances where a "person is not aware of a substantial risk that a death will *result* from his or her conduct, but should have been aware of such a risk."[71]

While national criminal law frameworks differ considerably, there are similarities regarding causation which are relevant here. Taking Germany as a representative example of civil law traditions, the *Strafgesetzbuch*

[66] See, on this topic, "Making Autonomous Vehicles", note 21 above.

[67] While the United States' Model Penal Code does not contain a provision dealing with vehicular homicide, legislations in certain domestic systems envisage it.

[68] See *State of Arizona*, note 4 above.

[69] American Law Institute, Model Penal Code: Official Draft and Explanatory Notes: Complete Text of Model Penal Code as Adopted at the 1962 Annual Meeting of the American Law Institute at Washington, DC, May 24, 1962 (Philadelphia, PA: American Law Institute, 1985) [Model Penal Code].

[70] Ibid., §2.13(1)(b); see Paul H. Robinson, "United States" in Kevin Jon Heller & Markus Dubber (eds.), *The Handbook of Comparative Criminal Law* (Redwood City, CA: Stanford University Press, 2011) ["United States"] 563 at 585 (emphasis added).

[71] Ibid. (emphasis added).

(German Criminal Code) (StGB) distinguishes two forms of intentional homicide: murder[72] and manslaughter.[73] Willingly taking the risk of causing death is sufficient for manslaughter.[74] Negligent homicide is proscribed separately,[75] and the *actus reus* consists of causing the death of a person through negligence.[76]

These are crimes of result, where the harm consists of the death of a person. While programmer conduct may be remote with regard to AV incidents, some decisions taken by AV programmers at an early stage of development could decisively impact the navigation behavior of an AV that results in a death. In other words, it is conceivable that a faulty algorithm designed by a programmer could cause a fatal road accident. The question then becomes what is the threshold of causal control exercised by programmers over an AV's unlawful behavior of navigation and its unlawful effects such as a human death.

IV.B Actus Reus *in AW-Related War Crimes*

This section addresses AW-related war crimes and whether programmers could perform the required *actus reus*. Since the *actus reus* would most likely stem from an AW's failure to distinguish between civilian and military targets, the war crime of indiscriminate attacks, which criminalizes violations of the aforementioned IHL rule of distinction,[77] takes on central importance.[78] The war crime of indiscriminate attacks refers inter alia to an attack that strikes military objectives and civilians or civilian objects without distinction. This can occur as a result of the use of weapons that are incapable of being directed at a specific military objective or accurately distinguishing between civilians and civilian

[72] *Strafgesetzbuch* (German Criminal Code), Germany (November 13, 1998 (Federal Law Gazette I, p. 3322), as amended by Art. 2 of the Act of June 19, 2019 (Federal Law Gazette I, p. 844)) [StGB], §211(1) (emphasis added).

[73] Under German criminal law, manslaughter is the intentional killing of another person without aggravating circumstances: StGB, note 72 above, §212.

[74] "Germany", note 13 above, at 262.

[75] StGB, note 72 above, §222.

[76] "Germany", note 13 above, at 263.

[77] For the underlying IHL, see AP I, note 17 above, Art. 51(4)(a); see also Jean-Marie Henckaerts & Louise Doswald-Beck, *Customary International Humanitarian Law, vol. 1: Rules* (New York, NY: Cambridge University Press, 2005), Rule 12, at 40.

[78] See Marta Bo, "Autonomous Weapons and the Responsibility Gap in Light of the Mens Rea of the War Crime of Attacking Civilians in the ICC Statute" (2021) 19:2 *Journal of International Criminal Justice* 275 ["Autonomous Weapons"] at 282–285.

objects and military objectives; these weapons are known as inherently indiscriminate weapons.[79]

While this war crime is neither specifically codified in the Rome Statute nor in AP I, it has been subsumed[80] under the war crime of directing attacks against civilians. Under AP I, the *actus reus* of the crime is defined in terms of causing death or injury.[81] In crimes of result with AWs, a causal nexus between the effects resulting from the deployment of an AW and a programmer's conduct must be established. Under the Rome Statute, the war crime is formulated as a conduct crime, proscribing the *actus reus* as the "directing of an attack" against civilians.[82] A causal nexus must be established between the unlawful AW's behavior and/or the attack and the programmer's conduct.[83] Under both frameworks, the question is whether programmers exercised causal control over the behavior and/or effects, e.g., death or attack, of an AW.

A final issue relates to the required nexus with an armed conflict. The Rome Statute requires that the conduct must take place "in the context of and was associated with" an armed conflict.[84] However, while undoubtedly there is a temporal and physical distance between programmer conduct and the armed conflict, it is conceivable that programmers may program AW software or upgrade it during an armed conflict. In certain instances, it could be argued that programmer control continues

[79] Knut Dörmann, *Elements of War Crimes under the Rome Statute of the International Criminal Court: Sources and Commentary* (Cambridge, UK: Cambridge University Press, 2003) [*Elements of War Crimes*] at 131 and 132; it is worth noting that programmers may have a greater role and responsibility, particularly when it comes to inherently indiscriminate weapons.

[80] Both by the ICC and the International Criminal Tribunal for the former Yugoslavia. The latter interpreted violations of Art. 3 of its Statute, relevant to unlawful attack charges, by resorting to AP I, note 17 above, Art. 85(3); See "Autonomous Weapons", note 78 above, at 283 and 284.

[81] AP I, note 17 above, Art. 85(3), the *actus reus* of the war crime of willfully launching attacks against civilians contains the requirement that an attack against civilians causes "death or serious injury to body or health."

[82] Rome Statute, note 16 above, Arts. 8(2)(b)(i) and 8(2)(e)(i).

[83] Moreover, under the Rome Statute, an attack could be considered as a result; Albin Eser, "Mental Elements – Mistake of Fact and Mistake of Law" in Antonio Cassese, Paola Gaeta, & John R.W.D. Jones (eds.), *The Rome Statute of the International Criminal Court: A Commentary* (New York, NY: Oxford University Press, 2002) 889 at 911.

[84] Element 4 of the elements of the crime at Rome Statute, note 16 above, Art. 8(2)(b)(i). As elaborated by the International Tribunal for the former Yugoslavia, the law of war crimes applies "from the initiation of … an armed conflict and extend beyond the cessation of hostilities until a general conclusion of peace is reached"; *Elements of War Crimes*, note 79 above, at 19–20.

even after the completion of the act of programming, when the effects of their decisions materialize in the behavior and/or effects of AWs in armed conflict. Programmers can be said to exercise a form of control over the behavior and/or effects of AWs that begins with the act of programming and continues thereafter.

IV.C The Causal Nexus between Programming and AV- and AW-Related Crimes

A crucial aspect of programmer criminal responsibility is the causal control they exercise over the behavior and/or effects of AVs and AWs. The assessment of causation refers to the conditions under which an AV's and AW's unlawful behavior and/or effects should be deemed the result of programmer conduct for purposes of holding them criminally responsible.

Causality is a complex topic. In common law and civil law countries, several tests to establish causation have been put forward. Due to difficulties in establishing a uniform test for causation, it has been argued that determining conditions for causation are "ultimately a matter of legal policy."[85] But this does not render the formulation of causality tests in the relevant criminal provisions completely beyond reach. While a comprehensive analysis of these theories is beyond the scope of this chapter, for the purposes of establishing when programmers exercise causal control, some theories are more aligned with the policy objectives pursued by the suppression of AV- and AW-related crimes.

First, in common law and civil law countries, the "but-for"/*conditio sine qua non* test is the dominant test for establishing physical causation, and it is intended as a relationship of physical cause and effect.[86] In the language of MPC §2.03(1)(a), the conduct must be "an antecedent but for which the result in question would not have occurred." The "but for" test works satisfactorily in cases of straightforward cause and effect, e.g., pointing a loaded gun toward the chest of another person and pulling the trigger. However, AV- and AW-related crimes are characterized by a temporal and physical gap between programmer conduct and the behavior

[85] "Causation in Criminal Law", note 18 above, at 785; see *contra Basic Concepts*, note 65 above, at 63 and 66.

[86] See "Causation in Criminal Law", note 18 above, at 787; also described as "empirical causality," which refers to the "metaphysical [and deterministic] question of cause and effect"; Marjolein Cupido, "Causation in International Crimes Cases: (Re)Conceptualizing the Causal Linkage" (2021) 32:1 *Criminal Law Forum* 1, ["International Crimes"] at 24.

and effect of AVs and AWs. They involve complex interactions between AVs and AWs and humans, including programmers, data providers and labelers, users, etc. AI itself is also a factor that could intervene in the causal chain. The problem of causation in these cases must thus be framed in a way that reflects the relevance of intervening and superseding causal forces which may break the causal nexus between a programmer's conduct and AV- and AW-related crime.

Both civil law and common law systems have adopted several theories to overcome the shortcomings[87] and correct the potential over-inclusiveness[88] of the "but-for" test, in complex cases involving numerous necessary conditions. Some of these theories include elements of foreseeability in the causality test.

The MPC adopts the "proximate cause test," which "differentiates among the many possible 'but for' causal forces, identifying some as 'necessary conditions' – necessary for the result to occur but not its direct 'cause' – and recognising others as the 'direct' or 'proximate' cause of the result."[89] The relationship is "direct" when the result is foreseeable and as such "this theory introduces an element of culpability into the law of causation."[90]

German theories about adequacy assert that whether a certain factor can be considered a cause of a certain effect depends on "whether conditions of that type do, generally, in the light of experience, produce effects of that nature."[91] These theories, which are not applied in their pure form in criminal law, include assessments that resemble a culpability assessment. They bring elements of foreseeability and culpability into the causality test, and in particular, a probability and possibility judgment regarding the actions of the accused.[92] However, these theories leave unresolved the different knowledge perspectives, i.e., objective, subjective, or mixed, on which the foreseeability assessment is to be based.[93]

Other causation theories include an element of understandability, awareness, or foreseeability of risks. In the MPC, the "harm-within-the risk" theory considers that causation in reckless and negligent crimes is

[87] "Causation in Criminal Law", note 18 above, at 787.

[88] Ibid.

[89] Arthur Leavens, "A Causation Approach to Criminal Omissions" (1988) 76 *California Law Review* 547 ["Causation Approach"] at 564.

[90] "Causation in Criminal Law", note 18 above, at 789.

[91] Ibid. at 791.

[92] Ibid. at 792.

[93] Ibid. at 795.

in principle established when the result was within the “risk of which the actor is aware or … of which he should be aware.”[94] In German criminal law, some theories describe causation in terms of the creation or aggravation of risk and limit causation to the unlawful risks that the violated criminal law provision intended to prevent.[95]

In response to the drawbacks of these theories, the teleological theory of causation holds that in all cases involving a so-called intervening independent causal force, the criterion should be whether the intervening causal force was “produced by ‘chance’ or was rather imputable to the criminal act in issue.”[96] Someone would be responsible for the result if their actions contributed in any manner to the intervening factor. What matters is the accused’s control over the criminal conduct and whether the intervening factor was connected in a but/for sense to their criminal act,[97] thus falling within their control.

In ICL, a conceptualization of causation that goes beyond the physical relation between acts and effects is more embryonic. However, it has been suggested that theories drawn from national criminal law systems, such as risk-taking and linking causation to culpability, and thus to foreseeability, should inform a theory of causation in ICL.[98] It has also been suggested that causality should entail an evaluation of the functional obligations of an actor and their area of operation in the economic sphere. According to this theory, causation is “connected to an individual’s control and scope of influence” and is limited to “dangers that he creates through his activity and has the power to avoid.”[99] As applied in the context of international crimes, which have a collective dimension, these theories could usefully be employed in the context of AV and AW development, which is collective by nature and is characterized by a distribution of responsibilities.

Programmers in some instances will cause harm through omission, notably by failing to avert a particular harmful risk when they are under a

[94] Model Penal Code, note 69 above, §2.03(3); §2.03(2) and (3) formulate several exceptions to the general proximity standard in cases of intervening and superseding causal forces.

[95] Among the “but-for” conditions that are *not* considered attributable are: “[a] consequence that the perpetrator has caused … if that act did not unjustifiably increase a risk”; “[a] consequence was not one to be averted by the rule the perpetrator violated”; and “if a voluntary act of risk taking on the part of the victim or a third person intervened.” For details, see “Germany”, note 13 above, at 268. See also “International Crimes”, note 86 above, at 26 and 27.

[96] “Causation in Criminal Law”, note 18 above, at 797.

[97] Ibid. at 798.

[98] “International Crimes”, note 86 above, at 43–47.

[99] “International Crimes”, note 86 above, at 41.

legal duty to prevent harmful events of that type ("commission by omission").[100] In these cases, the establishment of causation will be hypothetical as there is no physical cause-effect relationship between an omission and the proscribed result.[101] Other instances concern whether negligence on the side of the programmers, via, e.g., a lack of instructions and warnings, have contributed to and caused the omission, constituting a failure to intervene on behalf of the user. Such omissions amount to negligence, i.e., violations of positive duties of care,[102] and since it belongs to *mens rea*, will be addressed in the following section.

IV.D Criminal Negligence: Programming AVs and AWs

In light of the integration of culpability assessments in causation tests, an assessment of programmers' criminal responsibility would be incomplete without addressing *mens rea* issues. In relation to *mens rea*, while intentionally and knowingly programming an AV or AW to commit crimes falls squarely under these prohibitions, in both these contexts, the most expected and problematic issue is the unintended commission of these crimes, i.e., cases in which the programmer did not design the AI system to commit an offense, but harm nevertheless arises during its use.[103] In such situations, programmers had no intention to commit an offense, but still might incur criminal liability for risks that they should have known and foreseen. To define the scope of criminal responsibility for unintended harm, it is crucial to determine which risks can be known and foreseen by an AV or AW programmer.

There are important differences in the *mens rea* requirements of AV- and AW-related crimes. Under domestic criminal law, the standards of recklessness and negligence apply to the AV-related crimes of manslaughter and negligent homicide. While "[a] person acts 'recklessly' with regard to a result if he or she *consciously disregards a substantial risk* that his or

[100] StGB, note 72 above, §13.

[101] On causation in criminal omissions, see Graham Hughes, "Criminal Omissions" (1958) 67:4 *Yale Law Journal* 590 at 627–631. Causation in "commission by omission" is strictly connected with duties to act and duty to prevent a certain harm: see George Fletcher, *Rethinking Criminal Law* (New York, NY: Oxford University Press, 2000) at 606; "Causation Approach", note 89 above, at 562.

[102] See Marta Bo, "Criminal Responsibility by Omission for Failures to Stop Autonomous Weapon Systems" (2023) 21:5 *Journal of International Criminal Justice* 1057.

[103] See also Sabine Gless, Emily Silverman, & Thomas Weigend, "If Robots Cause Harm, Who Is to Blame? Self-Driving Cars and Criminal Liability" (2016) 19:3 *New Criminal Law Review* 412 at 425.

her conduct will cause the result; he or she acts only 'negligently' if he or she is *unaware of the substantial risk but should have perceived it*."[104] The MPC provides that "criminal homicide constitutes manslaughter when it is committed recklessly."[105] In the StGB, *dolus eventualis*, i.e., willingly taking the risk of causing death, would encompass situations covered by recklessness and is sufficient for manslaughter.[106] For negligent homicide,[107] one of the prerequisites is that the perpetrator can foresee the risk to a protected interest.[108]

Risk-based *mentes reae* are subject to more dispute in ICL. The International Tribunal for the former Yugoslavia accepted that recklessness could be a sufficient *mens rea* for the war crime of indiscriminate attacks under Article 85(3)(a) of AP I.[109] However, whether recklessness and *dolus eventualis* could be sufficient to ascribe criminal responsibility for war crimes within the framework of the Rome Statute remains debated.[110]

Unlike incidents with AVs, incidents in war resulting from a programmer's negligence cannot give rise to their criminal responsibility. Where applicable, recklessness and *dolus eventualis*, which entail understandability and foreseeability of risks of developing inherently indiscriminate AWs, become crucial to attribute responsibility to programmers in scenarios where programmers foresaw and took some risks. Excluding these mental elements would amount to ruling out the criminal responsibility of programmers in most expected instances of war crimes.

V Developing an International Criminal Law-Infused Notion of Meaningful Human Control over AVs and AWs that Incorporates *Mens Rea* and Causation Requirements

This section considers a notion of MHC applicable to AVs and AWs that is based on criminal law and that could function as a criminal

[104] "United States", note 70 above, at 575 (emphasis added); see also Guyora Binder, "Homicide" in Markus Dubber & Tatjana Hörnle (eds.), *The Oxford Handbook of Criminal Law* (New York, NY: Oxford University Press, 2014) 702 at 719: "Negligent manslaughter now usually requires objective foreseeability of death, rather than the simple violation of a duty of care."

[105] Model Penal Code, note 69 above, §2.13(1)(b).

[106] "Germany", note 13 above, at 262.

[107] StGB, note 72 above, §222.

[108] "Germany", note 13 above, at 263.

[109] See the case law quoted in "Autonomous Weapons", note 78 above, at 293.

[110] "Autonomous Weapons", note 78 above, at 286–294.

responsibility "anchor" or "attractor."[111] This is not the first attempt to develop a conception of control applicable to both AVs and AWs. Studies on MHC over AWs and moral responsibility of AWs[112] have been extended to AVs.[113] In their view, MHC should entail an element of traceability entailing that "*one human agent in the design history* or use context involved in designing, programming, operating and deploying the autonomous system … *understands or is in the position to understand the possible effects* in the world of the use of this system."[114] Traceability requires that someone in the design or use understands the capabilities of the AI system and its effects.

In line with these studies, it is argued here that programmers may decide and control how both traffic law and IHL are embedded in the respective algorithms, how AI systems see and move, and how they react to changes in the environment. McFarland and McCormack affirm that programmers may exercise control not only over an abstract range of behavior, but also in relation to specific behavior and effects of AWs.[115] Against this background, this chapter contends that programmer control begins at the initial stage of the AI development process and continues into the use phase, extending to the behavior and effects of AVs and AWs.

Assuming programmer control over certain AV- and AW-related unlawful behavior and effects, how can MHC be conceptualized so as to ensure that criminal responsibility is traced back to programmers when warranted? The foregoing discussion of causality in the context of AV- and AW-related crimes suggests that theories of causation that go beyond deterministic cause-and-effect assessments are particularly amenable to developing a theory of MHC that could ensure responsibility. These theories either link causation to *mens rea* standards or

[111] Daniele Amoroso & Guglielmo Tamburrini, "Autonomous Weapons Systems and Meaningful Human Control: Ethical and Legal Issues" (2020) 1 *Current Robotics Reports* 187 at 189.

[112] "MHC over Autonomous Systems", note 9 above, at 6–9.

[113] Simeon C. Calvert, Daniel Heikoop, Giulio Mecacci *et al.*, "A Human Centric Framework for the Analysis of Automated Driving Systems Based on Meaningful Human Control" (2020) 21:3 *Theoretical Issues in Ergonomics Science* 478 ["Human Centric Framework"] at 490–492.

[114] "MHC over Autonomous Systems", note 9 above, at 9; "Human Centric Framework", note 113 above, at 490 and 491 (emphasis added).

[115] Tim McFarland & Tim McCormack, "Mind the Gap: Can Developers of Autonomous Weapons Systems Be Liable for War Crimes?" (2014) 90 *International Law Studies* 361 at 366.

describe it in terms of the aggravation of risk. In either case, the ability to understand the capabilities of AI systems and their effects, and foreseeability of risks, are required. Considering these theories of causation in view of recent studies on MHC over AVs and AWs, the MHC's requirement of traceability arguably translates into the requirement of foreseeability of risks.[116] Because of the distribution of responsibilities in the context of AV and AW programming, causation theories introducing the notion of function-related risks are needed to limit programmers' criminal responsibility to those risks within their respective obligations and thus their sphere of influence and control. According to these theories, the risks that a programmer is obliged to prevent and that relate to their functional obligations, i.e., their function-related risks, could be considered causally imputable in principle.[117]

VI Conclusion

AVs and AWs are complex systems. Their programming implies a distribution of responsibilities and obligations within tech companies, and between them and manufacturers, third parties, and users, which makes it difficult to identify who may be responsible for harm stemming from their use. Despite the temporal and spatial gap between the programming phase and crimes, the responsibility of programmers in the commission of crimes should not be discarded. Indeed, crucial decisions on the behavior and effects of AVs and AWs are taken in the programming phase. While a more detailed case-by-case analysis is needed, this chapter has mapped out how programmers of AVs and AWs might be in control of certain AV- and AW-related risks and therefore criminally responsible for AV- and AW-related crimes.

This chapter has shown that the assessment of causation as a threshold for establishing whether an *actus reus* was committed may converge on the criteria of understandability and foreseeability of risks of unlawful behavior and/or effects of AVs and AWs. Those risks which fall within programmers' functional obligations and sphere of influence can be considered under their control and imputable to them.

[116] The anticipation of data issues is central to the above-mentioned UNIDIR report relating to data failures in AWs; see *Known Unknowns*, note 47 above, at 13 and 14.

[117] See Boutin and Woodcock arguing for the need to ensure MHC in the pre-deployment phase: "Realizing MHC", note 10 above.

Following this analysis, a notion of MHC applicable to programmers of AVs and AWs based on requirements for the imputation of criminal responsibility can be developed. It may function as a responsibility anchor in so far as it helps trace back responsibility to the individuals that could understand and foresee the risk of a crime being committed with an AV or AW.

MAMAK ©
154
97
48