# Coding Guide for the Military Videogame Advertisement Dataset

Marcus Schulzke (University of York) and Nick Robinson (University of Leeds)

Starting with a central research question, ‘how do we capture what we see and experience in the promotional advertisements for military videogames?', the 520 videos within our codebook have been watched on social media over 1 billion times.

The thinking which underpins the development of our codebook combines both qualitative and quantitative sensibilities. In its construction, it was heavily influenced by the qualitative literature on visual politics, especially work on militarism and militarization. That research helped to identify what qualities previous researchers considered to be signifiers of militarism (some visual and some not), thereby linking the dataset to existing research and making it ideal for both hypothesis testing and for reaching novel discoveries within the scope of ongoing debates on militarism and militarization. The data in the Military Videogame Advertisement (MVA) dataset was created through a process of coding visual information – transforming certain qualitative visual signifiers of militarism into numbers according to categories that were developed to identify theoretically significant representational patterns. The data is also heavily supplemented by descriptive variables. Some of these are intended to clarify or explain the coding decisions, while other descriptors capture information that was not amenable to quantification. The descriptors also allow for reflection on the affective impact of sound and visuals.

The coding process began with an effort to clearly define the universe of analysis. We set out to collect and code all videos that were hosted on YouTube by developers of the top fifty best-selling military videogames and that were clearly associated with those games. The top fifty games were identified using sales data that was posted on [www.vgchartz.com](http://www.vgchartz.com) (a widely recognized source for such data). We then identified the developers of those games and searched YouTube for the official accounts associated with those developers. In total this resulted in us coding a total of 520 videos.

All of the top selling games which had no presence on YouTube (principally because the games actually predated the advent of YouTube) were omitted from the data collection. We also avoided downloading and analysis of any videos that were produced by third-parties, as these videos did not necessarily reflect the marketing strategies of the companies that are involved in producing military videogames or they might have been edited in some way. After collecting all of the videos for each of the games, we then elected to exclude all of those which were over 10 minutes in length from the dataset. These videos (which amounted to just 9 of the total) were typically much longer than ten minutes, were highly atypical in form, and usually showed multiplayer tournaments. It was necessary to omit these from coding because their length made them prohibitively difficult to code and, even more importantly, because these atypical videos would skew the results. The typical length of those in the dataset being 2 minutes (128 seconds).

The coding procedure was to watch each video once to get a general sense of its content, then to watch it on multiple occasions while paying attention to specific variables. Each variable required at least one viewing, though sometimes two or three were necessary. Moreover, the variables that required the counting of visual signifiers, such as the numbers of male or female characters shown, could often only be coded by watching the video at a reduced speed.

The dataset includes five types of variables. *Descriptive variables* are text boxes that can be filled with any combination of numerical or textual information. These variables provide context for the coding decisions and make it possible to record visual or auditory experiences that were not amenable to capture by numbers or that fell outside the scope of the existing variables. *Ordinal variables* identify a cluster of related visual signifiers, such as the types of civilian structures that are shown in a video. The numbers assigned to these signifiers are arbitrary and only meaningful insofar as they reflect a particular visual pattern. *Binary variables* are coded as 0 or 1 to indicate the presence or absence of a particular signifier. For example, a video may be coded as '0' for 'Civilian Structures' if no civilian structures are evident in the video or as '1' if civilian structures are shown. When it was impossible to determine whether a signifier was present or absent, these variables are coded -99 to reflect the uncertainty. *Interval variables* indicate the number of something that was counted in a video, such as the number of male characters that were shown. Unlike the numbers assigned to categorical variables, these numbers are not arbitrary. Finally, *confidence variables* indicate the coder's confidence that the value in the preceding column is accurate. These are always on a three-point scale ranging from 3 (most confident) to 1 (least confident). For example, a confidence value of 1 following the variable ‘Total Number of Characters’ would indicate that the coder has a low level of certainty that the number given is exactly right. This would suggest that the character count is an approximation and that the actual figure may be higher or lower than the recorded number by around 20%. Finally, as is exemplified by best practice in terms of the compilation of a codebook such as this, all of the coders initial coding was checked and sampled on the basis of an inter-coder reliability check.

The video ‘Official Call of Duty Advanced Warfare Reveal Trailer’ (<https://www.youtube.com/watch?v=sFu5qXMuaJU)> offers a prime example of how the coding process works when it is applied to military videogame advertisements. When it comes to the categorical variable 'Video Type' the video was coded 1 to reflect the fact that it was explicitly framed as a trailer and that it had the visual characteristics indicative of that type of video (such as the fast pace, overview of the game's story, and video footage drawn from multiple different parts of the game). The *binary variable* ‘Real Conflict’ was coded 0 to indicate that the video was not explicitly about any real historical or ongoing conflict. That is to say, no real conflicts were mentioned during the video, and the video indicates that the fighting is set sometime in the future. The *interval variable* 'Number of Characters' was coded as '125' to indicate that this was the approximate number of characters shown in the video. And the following *confidence* number '1' indicates a low level of certainty that this number is accurate. This *confidence number* was warranted because the game's fast pace makes it difficult to count the number of characters that were shown. The codebook also includes a wide variety of other variables such as ‘actor’ variables which differentiate between types of combatants, ‘military equipment’ variables which cover weapons and equipment featured in the video, ‘setting’ variables which cover geographical location depicted, and ‘conflict type’ variables which help identify the nature of the conflict and whether or not it is based on a ‘real’ conflict.

The many other variables of each type followed this same pattern, with the coder making judgments based on how the coding rules set out by each variable could be used to capture elements of the visual experience.

**Funding:**

This work was supported by the Swedish Research Council as part of its programme, ‘The Digitized Society - Past, Present, and Future’. Grant title: Jackson, S., Robinson, N., Schneiker, A., Joachim, J. (2012–16) ‘Militarization 2.0: militarization’s social media footprint through a gendered lens’ [grant number 340-2012-5990].

This work was also supported by the ‘University of Leeds Undergraduate Research & Leadership Scholarships Scheme (Laidlaw Scholarships), 2015-16’ which provided financial assistance and training for a student to work with us on the inter-coder reliability checks.