

Article

On Metrics for Location-Aware Games

Luis E. Rodríguez-Pupo *, Sven Casteleyn and Carlos Granell

Geospatial Technologies Research Group (GEOTEC), Universitat Jaume I, Av. Vicente Sos Baynat s/n, 12071 Castellón de la Plana, Spain; sven.casteleyn@uji.es (S.C.); carlos.granell@uji.es (C.G.)

* Correspondence: luis.rodriguez@uji.es

Received: 9 August 2017; Accepted: 25 September 2017; Published: 27 September 2017

Abstract: Metrics are important and well-known tools to measure users' behavior in games, and gameplay in general. Particularities of location-aware games—a class of games where the player's location plays a central role—demand specific support in metrics to adequately address the spatio-temporal features such games exhibit. In this article, we analyse and discuss how existing game analytics platforms address the spatio-temporal features of location-aware games. Our analysis reveals that little support is available. Next, based on the analysis, we propose a classification of spatial metrics, embedded in existing literature, and discuss three types of spatial metrics—point-, trajectory- and area-based metrics—and elaborate examples and difficulties. Finally, we discuss how spatial metrics may be deployed to improve gameplay in location-aware games.

Keywords: location-aware games; location-based games; geo-games; game analytics; spatio-temporal metrics; game analytics platforms

1. Introduction

Within the varied landscape of games genres and applications [1–6], location-aware games are games that take place in the real world space, using the location as an important aspect of the game, and often, but not always, require the user to move physically in space. Since the first variants of digitally-enabled treasure hunting games, such as geo-caching [6,7], location-aware games are gaining momentum and popularity. This culminated in the absolute success story of Pokemon Go, which has/had an unprecedented usage and user community [8]. In the literature, a wide variety of terminology is used to denote location-aware games, or slightly more specific variants of them, such as location-based games [9], pervasive or urban games [6], or geo-games [10], just to name a few. We discuss definitions, terminology, and differences in terminology in detail in Section 2. In this article, we use the term location-aware games, as most general term denoting games whose mechanics and rules are tightly coupled with the location, surrounding context, and/or geography. *Where* these games are being deployed and played clearly matters; the *where* may even turn out to be the most influential factor for the success or failure of location-aware games.

As for regular games, an important success factor for location-aware games is their playability. González Sánchez et al. defined playability as “a set of properties that describe the player experience using a specific game system whose main objective is to provide enjoyment and entertainment, by being credible and satisfying, when the player plays alone or in company” ([11], p. 67). This multifaceted concept involves usability, user experience and satisfaction, and the ability for users to achieve the game goals. As such, one of the measures of playability is game balance. Keeping a game “balanced” involves finding the right trade-off between important dimensions of a game, such as difficulty, challenges, progress, and incentives, which is ultimately reflected in the level of user engagement with the game. Game balance is especially challenging for game designers as games are getting more complex and sophisticated, both in a technical and narrative (i.e., storytelling) way. Therefore, it requires constant re-evaluation and follow up. Existing research

has therefore focused on the evaluation of game balance. Jaffe et al. [12] explored assessment methods for games in general, while Kiefer and Matyas [13] put the focus on spatio-temporal design parameters involved in games. Central to the previous and other methods for assessing game balance is to (continuously, on a regular basis or on demand) capture data related to relevant aspects and features of the game. For example, the movement of a player in the game scenario, the players interactions and their characteristics, and game rewards achieved by players can help to investigate and detect potential imbalances and deficiencies in the game. Game data collection is not only relevant at the design stage, but most importantly while the game is being “played” (deployed), to allow game designers to investigate and monitor those game aspects and dimensions that are difficult to assess during design time, such as user engagement level, among others.

The amount of data collected in games can be potentially big, depending on how many aspects are to be tracked and the number of users/players playing. For example, monitoring progress, access to resources, and players’ trajectories are some aspects in which game developers might be interested. In order to normalize and make it comparable, gathered data is processed by means of metrics (or game metrics). The term metric has been broadly used in different contexts and fields, such as urban studies [14], software engineering [15], ecology [16], and human computer interaction [17,18], but in all cases, the rationale behind metrics boils down to establishing a standard way of or common practice to describe certain relevant (high level) parameters of the game to be monitored, and allow comparison of the collected data.

The interplay between metrics and location-aware games is the main focus of the paper. It is worthwhile to note that this research is not about game metrics in general (e.g., [19]). As location-aware games are tightly-coupled to spatio-temporal dimensions, metrics for location-aware games will necessarily handle spatio-temporal characteristics of monitored data. Thus, geospatial data needs to be effectively collected, analyzed and computed. Spatial metrics have received little attention so far in the literature (with outstanding exceptions [20]); this article fills this gap in literature.

After an overview of the terminology and related concepts in Section 2, in order to establish a common understanding about location-aware games and metrics, the main contributions of the paper are covered in Sections 3 and 4:

- In Section 3, we analyze and compare the spatio-temporal support in data handling and metric analysis for different existing commercial and academic game analytics platforms and tools, and their application to location-aware games. The comparative analysis evaluates whether such platforms offer the required features for collecting, processing and visualizing spatio-temporal data. Next, we discuss current limitations of these platforms and tools to set the basis for defining an analytics platform for location-aware games.
- In Section 4, we propose a classification of spatially-related metrics, framed in the realm of other non-spatial metric categorizations from the literature. Our intent is to establish the desired spatial capabilities, which would allow game developers to perform analysis that could otherwise not be done with non-spatial game analytics.

Finally, we provide a discussion in Section 5 and conclude the article with our final conclusion and roadmap for further work in Section 6. While the scope of this article falls within the two aforementioned contributions, our mid-term research objective is to set out a conceptual platform to enable metrics definition and execution for location-aware games in terms of components, services and required functionality, accompanied with the implementation of an operational analytical platform to support the definition, monitoring and computation of metrics for location-aware games.

2. Definitions and Terminology

In this section, we introduce related terms and definitions and set out a common ground for subsequent sections.

2.1. Gameplay

Gameplay is a popular term used throughout the industry [21], which “typically refers to the behavior of a game (e.g., the rules, difficulty, consistency, fairness, goals, and constraints), rather than the presentation of the game (e.g., graphics, artwork, and sound)” ([22], p. 123). The role of gameplay is well known and, as such, game designers have put a considerable effort in designing, testing and refining gameplay. Quality assurance and playtesting, a testing technique during the game design process where players (testers) are asked to “think aloud” about their perception of different game aspects, are crucial parts of the design and development process. The evaluation of gameplay is a continuous process that leads to several cycles of adjustment and refinement during a game’s life-cycle.

Prensky takes a motivational viewpoint in that gameplay is “all the doing, thinking and decision making that makes a game, either fun, or not. In a puzzle game, the gameplay is the physical and mental activities in the puzzles. In a shooter, it is the player’s and the opponents’ speed and abilities. In a strategy game it is the available options and tactics” ([23], p. 9). For the author gameplay is also a dynamic, evolving concept that needs to be continuously considered as it includes “not only providing engaging activities, characters, and situation but also balancing and constantly adjusting the game so that it continually keeps the player in the ‘flow zone’” ([23], p. 9).

Other authors took complementary views of gameplay. For example, game designers [24] defined gameplay as “one or more causally linked series of challenges in a simulated environment”. This definition suggests that gameplay also includes the actions that players take to address the challenges and accomplish the game goals. Costkian [25] focused on the user/player perception by stating that “a good gameplay keeps a player motivated and engaged throughout an entire game”.

From the aforementioned, and as acknowledged by Kiili [26], gameplay lacks a precise definition among the gaming community. It seems so broad that related literature works have addressed it from distinct viewpoints and perspectives. Nevertheless, there is a common belief of the importance of gameplay (or overall game experience), and it is undeniable that “its significance should not be underestimate” [26]. Typically, gameplay includes the game’s goals and how to achieve them, the time to complete part(s) of the game, the “cost” (e.g., health, game money, resources), and the probability of succeeding. It is considered essential for the game to be “fun” and “enjoyable”, and the primary concern for game developers is to improve gameplay. As such, they need to measure and monitor those aspects and features of interest of the game, in order to improve gameplay as a whole. The focus of this article—within the context of metrics for location-aware games—is precisely on those aspects/features of gameplay that either have spatio-temporal connotations or are suitable to be analyzed using spatio-temporal methods and techniques (as part of metric analysis).

2.2. Location-Based Games, Location-Aware Games and Geo-Games

Schlieder et al. [9] defined location-based games as those that “involve body movements beyond figural space—that is, beyond the space of computer screens and small 3D objects”. The authors highlighted the importance of locomotion or movement “in vista space, typically a single room or sports field, or in environmental space, such as a neighborhood or city” (Montello [27] defines vista space as “space that can be visually apprehended from a single place without appreciable locomotion and environmental space as space too large and otherwise obscured to visually apprehend without considerable locomotion”), where the game takes place. Other authors define location-based games differently [28,29]. For them, these games do not necessarily require movement; they just need to take into account the user’s location and/or environment. Anticipating our argumentation line, we

envision a more exclusive definition of location-based games where user's movement is an important but not mandatory aspect of the game.

Kiefer and Matyas [13] distinguished between location-based games and geo-games, where the latter are a specific case of the former. According to the authors, in geo-games "a fixed number of players move between a fixed number of locations taking up and putting down resources when they reach a new location. Resources cannot move around without any involvement of a player, which is one basic constraint for geo-games". The authors call this constraint "spatial coherence", and declare it a defining characteristic of geo-games. Apart from the spatial coherence, Schlieder et al. [10] identified temporal coherence as a second constraint in geo-games. Temporal coherence asserts that performing an action needs time at least as long as the synchronization interval, a predefined constant for the game. Looking at the particular context of this definition though, the authors were introducing a framework for location-based games covering board games. This context (board games) explains the (perhaps overly) restrictive definition (i.e., fixed amount of players, focus on picking up or dropping resources, resources that can only be moved by players, turn-based games). While a general appreciation of spatial and temporal coherence as part of the game mechanics makes sense, we do not share the focused view and narrow definition of geo-games, which we think should be open to a larger array of location-based games besides (geospatial) boardgames.

To avoid confusion with the restrictions associated, at least by some authors, with location-based and geo-games, we use the term location-aware games in the article, going beyond of players' and resources' position and/or movement to cover an ample range of games where the game, and the players are *aware* of their location and (possibly) surroundings (i.e., their environment and the objects in it), and vice versa, the surrounding environment is *aware* of players. This broader definition includes any games where location plays a role, and thus includes location-based and geo-games, as well as other games where location (and environment) is important, such as pervasive or urban games [6]. In the remainder of this article, we embrace the term location-aware game, unless the contrary is explicitly stated.

2.3. Metrics in Location-Aware Games

In its mathematical interpretation, the term "metric" is equivalent to "a nonnegative function describing the distance between neighboring points for a given set" [30]. Regardless of the target domain or discipline, the recurrent and invariable aspect of a metric is that is a function, i.e., it processes gathered data to produce comparable results (indicators) for evaluation, monitoring and decision-making processes. Applied to the gaming context, we find the following definitions in literature (Table 1):

Table 1. Key aspects of metric definitions extracted from the literature.

| Authors | Metrics |
|--------------------------|--|
| Tychsen and Canossa [31] | "are numerical data obtained from user interaction with games", "denote a standard unit of measure", and "are utilized for quantitatively measuring and evaluating processes". |
| Drachen and Canossa [32] | "are instrumentation data about the user behavior and user-game interaction", and "provide detailed quantitative information about the player (user) behavior". |
| Hochleitner et al. [33] | "are interpretable measures of something, whereas telemetry is the raw data", and "represent telemetry data that have been transformed somehow". |
| Fürnkranz [21] | "are the aspects of gameplay of interest to the designer". |
| Medler et al. [34] | "are monitoring player behavior (e.g., logging in-game events)". |
| Kaner and Bond [35] | "are measurement functions". |
| Stanton et al. [36] | "are distance functions". |

Table 1 shows a mixture of interpretations of what a metric is. In game user research and games telemetry [21,31–34], a metric is termed both as “data used in” and as “data collected for evaluating” an aspect of the game. As such, the term metric is equated to data collected or data obtained, suggesting that metric (as a function) and (input or output) metrical data are used interchangeably. A possible reason might be that in some cases, raw acquired data is sufficient as a comparable metric, and further data analysis and computation is not required. In such cases, it may be difficult to distinguish between metric and metrical data.

To avoid aforementioned terminological inaccuracies, and establish and delimiting the scope and understanding of what a metric is in the context of location-aware games, we first provide clarity on what we understand under “metric”. We consider the most general definition for metrics is the mathematical definition, which is coherent with the one in the software engineering field [35]. This definition reflects the fact that collected (metrical) data is raw information, which in most cases is used for further evaluation rather than being the metric itself. To clarify, consider as an illustrative example a metric that provides a ratio of values, e.g., amount of movements per minute. Clearly, such a composite metric needs to be calculated from basic data, i.e., movement derivation from singular sensor values, temporal average computation. Notice that data is dynamic but re-usable (e.g., sensor data), whereas metrics are calculated for a specific purpose (e.g., amount of movements per minute, amount of lateral movements) and often static (i.e., execution of the same function over and over). Furthermore, normalization of data among sessions, or even among games, through metrics is what makes comparison of data possible.

Metrics are related to gameplay in the fact that they serve to measure certain parameters of the game that are considered to, at least partly, assess gameplay. While re-usable metrics are conceivable (e.g., total distance covered), it is unrealistic to think of a set of predefined metrics for measuring and assessing any facet of gameplay, in any game. Especially in location-aware games, where location and spatial context may notably influence which aspects of gameplay are going to be monitored and how. Consequently, the ability to support the design and creation of custom metrics is a quintessential feature for game analysts, designers and developers, and a clear separation of concerns between metrics and metrical data is therefore essential to allow game developers the degree of flexibility and customization required for metric design and creation. In this article, we focus on metrics, and associated geospatial data, which are relevant for games with a spatio-temporal dimension; in other words, metrics for location-aware games.

With the concepts of “metrics” and “metrical data” clarified, and our focus on their spatio-temporal dimension outlined, we subsequently study the support in existing commercial and academic platforms for geospatial game metrics and analysis.

3. Geospatial Support in Current Games Analytics Platforms and Tools

Given the importance of measuring different aspects of games during design, development and operational phases, several companies and research institutions have invested significant resources in platforms and tools for metrical data collection and analysis. Nevertheless, these commercial and academic tools are strongly influenced by specific purposes and aims, or are general in scope and cover only main scenarios and use cases. In either case, none of the surveyed tools were specifically designed for location-aware games, that is, spatial and/or spatio-temporal features were not natively included even though some support and workarounds may exist. To the best of our knowledge, there is no tool or platform to specifically handle metrics for location-aware games. As mentioned earlier, this is a mid-term goal of our research and this article contributes to the path to accomplish it. In this section, we conduct a comparative analysis of popular games analytics tools to examine what is currently covered and, next, discuss limitations and missing features in the realm of location-aware games.

3.1. Background for Tool Comparison

The comparative analysis allowed us to identify a set of commonly available features to ideally support game analysts and developers to be able to compute metrics. While analyzing the tools, we identified these features and grouped them into four functional areas, which schematically form a conceptual architecture shown in Figure 1 that frames the process for tool comparison.

Typically, most of the surveyed platforms and tools follow a client-server architecture, in which the game clients collect raw data and transmit it to the server platform for data storage and processing (i.e., metric computation). In the context of games analytics, a client-server architecture eases computation and further access to data by keeping data centralized and accessible through an abstraction layer, which typically takes the form of service endpoints or data access APIs. Such centralized data availability is particularly important in the case of multi-player systems, where data synchronization among multiple players needs to be supported. Additional reporting and visualization of processed data (metric outputs) are usually developed as extensions or add-on components to the analytical platform. The general conceptual architecture shown in Figure 1, which is not yet intended to hypothesize an operational analytical platform for geospatial metric processing, helps us to delimit the functional areas of existing and desirable features for the comparative analysis.

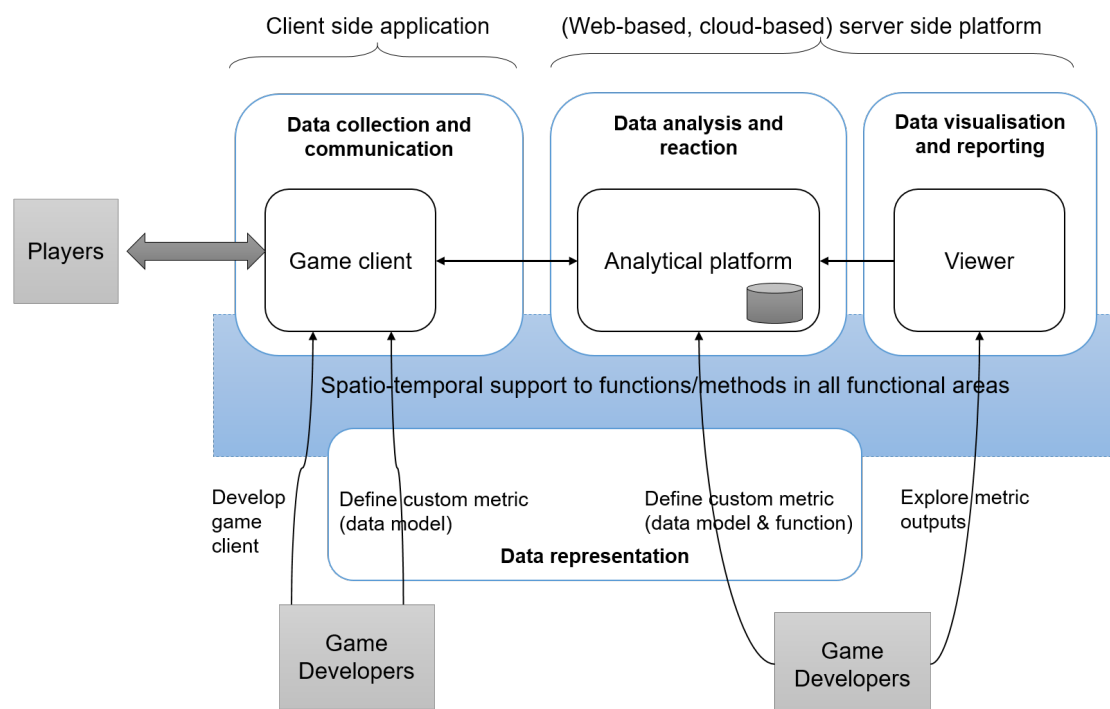


Figure 1. Schematic architecture for (location-aware) games analytical platforms and tools.

The first functional area is concerned with data collection and communication. Both are basic operations of game clients, which collect raw data for monitoring and tracking user interactions and behavior with(in) the game and transfer it (involving data transformation and wrangling if required) to the analytical platform. These two actions together are often termed as telemetry in the game community [37], whereas the GIS community tends to use single terms: data collection (or data gathering) plus data transferring. Game developers implement game clients coupled to specific game engines (e.g., Unity) or operating systems (e.g., Android, iOS) using custom SDKs (Software Development Kit) associated with the analytical platforms or tools.

The second functional area, data representation, is shared between the client and server side developments of analytics platforms. It deals with the data model to structure the monitored

phenomena of the game. Game clients capture raw data and instantiate data models accordingly. Analytical platforms compute metrics based in these shared data models, either as inputs or outputs.

The third functional area, data analysis and reaction, resides at the server side; they are the main tasks of a game analytical platform. The term analytic refers to metrics computation i.e., the execution of distance functions over collected data. Reaction is concerned with the ability of the platform to *react* when certain conditions are met. Examples of reactions are callback and notification mechanisms, or programmed actions that are being triggered subject to metric outputs.

Finally, data visualization and reporting are the focus of the fourth functional area. Game developers are often provided with monitoring and visualization tools to explore processed data in varied forms (visual, table-form etc.) in order to make informed decisions to improve overall gameplay.

All features of the surveyed tools and platforms fall within one of these four functional areas. Beyond feature categorization, these functional areas can be considered as main building blocks for conceptually defining an operational platform for metric computation for location-aware games, in which spatial and/or spatio-temporal support is a cross-cutting layer, thereby ensuring support for handling spatial and/or spatio-temporal data and techniques in any of these functional areas.

3.2. Comparative Analysis

In the following Tables, Table 2 summarises the relation between functional areas and their features. In Tables 3–7, we compare popular platforms and tools for game metric analytics, ordered by functional area. Data representation is shared by the client- and server-side functional areas.

Table 2. Functional areas and contained features for game metrics.

| Functional Areas | Features |
|-----------------------------------|--|
| Data collection and communication | <ul style="list-style-type: none"> • Data collection strategies • Data communication strategies • Client-side development support |
| Data representation | <ul style="list-style-type: none"> • Default event (data model) types • Custom event types definition • Spatial support (for data modelling) |
| Data analysis and reaction | <p><i>Analysis</i></p> <ul style="list-style-type: none"> • Default metrics and games analysis • Custom metrics definition • Spatial support (for analysis) <p><i>Reaction</i></p> <ul style="list-style-type: none"> • Reactive rules • Spatial support (conditions & actions) |
| Data visualisation and reporting | <ul style="list-style-type: none"> • (Open) Data Access • Visual analytics • Spatial support (for visual analytics) |

With respect to the data collection and communication functional area, the supported features encountered in the surveyed tools are described next, and Table 3 summarizes the type of support for data collection and communication per feature:

- **Data collection.** Strategies for data collection can be either time-based or event-based [38]. The differentiation between the two is mainly determined by the fact that data associated with the former strategy is produced at high rates or does not exhibit high variabilities over short periods

of time, while on the contrary, data required for event triggering is usually collected and produced “eventually”, based on a certain frequency or as a result of an action in the game. The latter is therefore more meaningful to game developers in the context of the game. An example of time-based data collection is capturing user location at a certain frequency, while an example of event-based data collection is the usage of a given weapon, which can only be collected in case the player changes and selects the particular weapon. Game analytics platforms usually do not distinguish between time- and event-based data collection, which is considered a responsibility of the game developer; they only provide facilities for defining event types and communicating captured data (see further on), independent of their data collection strategy. Indeed, all reviewed game analytical tools follow this approach. We thus do not include data collection strategies in Table 3. Nevertheless, the way data is collected may influence how it can be processed, and therefore influence the choice of analytical methods. We visit this later on in Section 3.3.

Another way of qualifying data collection is based on *who*, rather than *how*. Drachen and Schubert [39] categorize collected data into player-derived data and system-derived data. Player-derived data refers to data that, at least partly, captures (some) player behavior. Four dimensions are hereby considered relevant: *who*, *what*, *when* and *where*. Examples of player-derived data include players’ items owned, position, trajectories, etc. We go into more detail on the spatial dimensions of player-derived data when discussing the data representation functional area further on in this section. System-derived data refers to data generated by the game, and is useful for e.g., monitoring technical issues (e.g., network balancing, bug tracking) or the game as a whole. Game analytics engines do not distinguish between player- and system-derived data; it is the responsibility of the game developer to define appropriate event types and communicate the relevant data to track.

- **Data communication.** It indicates how gathered data is imported or fed into the analytical tool. As mentioned earlier, most of the surveyed tools follow a common architectural pattern: a client part collects data, and a server part performs analytical tasks and provides services for getting the collected data and store it permanently into repositories. Mechanisms for data communication, i.e, how gathered data is transferred from the client to server, can be roughly classified in two groups: data streaming or data upload. In case of data streaming, events are immediately streamed to the server, as real-time reflection of events in the game is important (e.g., real-time game strategy) and overall system performance is not relatively affected by the frequency of data transmission. It has the advantage of enabling on-the-fly analysis over the streamed data.

On the other hand, in the data upload group, bits of data are gathered over a certain (usually short) period of time, packaged together, and sent to the server. It is useful in cases where events occur at high rates and/or involve large amounts of data, and therefore immediate and frequent unitary communication might disrupt system performance. On the downside, the lack of live streamed data limits the possibility for real-time data analysis. Indeed, this approach mainly serves analysis *a posteriori*, and often includes the use of log files (or databases), and batch bulk updates to a server platform for non-real-time analysis. The main reason for using this strategy for data communication is that the amount of data produced could lead to undesirable levels of network load or decreased performance.

- **Client-side development support.** Supporting tools for game development facilitate the integration of data communication protocols in game clients. This is essential to free game developers from knowing the particularities of the ways to transfer collected data to the analytics back-end platform. Commercial tools mostly provide software development kits (SDKs) for target mobile operating systems (e.g., iOS, Windows, Android) and game engines (e.g., Unity, GameMaker Studio, etc.). REST Application Programming Interfaces (APIs) are another way to support game development, which are supported in varied programming languages

(e.g., JavaScript, C#, Java). In general, SDKs and REST APIs are often viewed as secondary features but are extremely useful to make developers' life easier for, e.g., being in compliance with the protocol to validating and submitting collected data.

Table 3. Comparison of data collection and communication related features among academic and commercial tools for game analytics. NA/NS = Not Applicable, Not Specified or Unknown.

| Tool/Citation | Data Communication | Client-Side Development Support |
|--------------------|--------------------|--|
| WebTics [40] | Streamed events | HTTP clients |
| TRUE [41] | Data Upload | NA/NS |
| DataCracker [34] | Streamed events | NA/NS |
| Skynet [42] | Streamed events | NA/NS |
| GameAnalytics [43] | Streamed events | SDKs for Android, iOS, Xamarin, Unreal Engine, Unity; REST API |
| HoneyTracks [44] | Streamed events | SDKs for Android, iOS, Unity C#; REST API |
| Xsolla [45] | Data Upload | HTTP |
| GameGuts [46] | Streamed events | Java |
| DeltaDNA [47] | Streamed events | SDKs Unity, Android, iOS, GameMaker Studio; REST API |

Table 4 summarises the type of support for the functional area of data representation in the surveyed tools, according to the following features:

- **Default event types.** Most surveyed platforms include the definitions of events as logical data structures (data models), called event types. Some of them provide default event types, of which each has an associated data model and often, but not necessarily, related processing methods and/or predefined metrics. For example, GameAnalytics [43] provides a “business” event type that is used to track (and validate) real-money transactions in games. Based on this business event type, GameAnalytics supports various types of revenue metrics, such as average daily revenue per daily active user or per paying user.

It is worth noting that the way game analytics platforms offer event types influences their extensibility and customizability. The most common logical data model for event types are records, with a fixed number of fields that represent the monitored phenomenon. Record-based event types are useful in many cases, where the data model is relatively static, and a number of event types are predefined for being used in previously known metrics. Another commonly used logical data model is based on arrays of key-value pairs (e.g., DataCracker [34]), which are more versatile and customizable data models than those based on data records mentioned above. From the surveyed game analytics platforms, [40,42–44] provide record-based; [34,41,46,47] provide key-value pairs (We consider structured documents like XML (as in the case of [41]), or even unspecified structures that are dynamic (in commercial systems sometimes it is not documented), to be similar to arrays of key-value pairs for the purpose of this comparison). like event types; and [45] provides a combination of both.

- **Custom event types.** Contrary to default event types, custom event types allow game developers to define their own data model to handle any type of monitored aspect of the game. Custom event types provide high-level support for game-specific events, and the (composite) data they require. This feature is linked to the definition of custom metrics (see the third functional area below), as the combination of both gives game developers the freedom to extend the capabilities of a game analytics platform to practically monitor and compute any facet of a game. The support for this feature is varied among the analysed platforms, although most of them restrict the creation of new data models to those based exclusively on simple data types (i.e., boolean, string, integer). Support for more complex and nested data structures was practically inexistent. Reasons might be the absence of a flexible processing model and/or rigid storage schemes to handle data structures beyond that of a sequence of simple data types.
- **Geospatial data support.** As mentioned earlier, Drachen and Schubert [39] categorised collected data for game analytics into player-derived data and system-derived data. From the four

dimensions for player-derived data (who, what, where, when), two are particularly relevant from the spatio-temporal point of view: “where is *this* happening?” and “when (at what time) is *that* happening?”, where this/that is any relevant (player-specific) event in the game. For the former (where), the authors pointed to “the spatial position, user location, movement, speed, orientation”. In case of games where no avatars exist, location may refer to the screen coordinates when users touch or swipe on the gaming device. The latter (when) looks for a temporal reference, which may refer to actual or in-game time, point in time, time interval, timespan, etc. As space and time are so tightly integrated in games, custom data models for event types should ideally be able to natively manage and store data structures to support the aforementioned spatio-temporal characteristics.

That is why, specifically for the purpose of this paper on the study of geospatial metrics for location-aware games, we envision the support for geospatial data as a cross-cutting layer (see Figure 1), i.e., where spatio-temporal and/or spatial characteristics of collected data are supported, either by default or by custom event types. In practice, though, native support is scarce, and game developers are mostly limited to create simple representation of space: pairs of coordinates. To do so, we found two approaches in between the tools that provide some support for capturing geo-spatial data. A first group of tools ([34,40–42,47]) is able to capture coordinates as a pair of simple attributes (location/space attributes), which is obviously quite far from being able to really handle geospatial data. A second group ([43,45,46]) does not provide explicit support, but is possible to store the coordinates by mean of other mechanisms, usually custom attributes. In these cases it is up to the developer to implement the best way to store spatio-temporal information.

Table 4. Comparison of data representation related features among academic and commercial tools for game analytics. NA/NS=Not Applicable, Not Specified or Unknown.

| Tool/Citation | Default Event Types | Custom Event Types | Geospatial Data Support |
|--------------------|--|--------------------|---------------------------|
| WebTics [40] | NA/NS | Yes(simple) | Location/space attributes |
| TRUE [41] | NA/NS | Yes | Location/space attributes |
| DataCracker [34] | NA/NS | Yes (simple) | Location/space attributes |
| Skynet [42] | NA/NS | Yes | Location/space attributes |
| GameAnalytics [43] | Resource, Business, Progression, Error | Yes (simple) | via custom attributes |
| HoneyTracks [44] | User, Session, Business, Social | Yes (simple) | NA/NS |
| Xsolla [45] | Business, User, System | Yes (simple) | via custom attributes |
| GameGuts [46] | NA/NS | Yes (simple-DSL) | via custom attributes |
| DeltaDNA [47] | Many types | Yes (simple) | Location/space attributes |

For the third main functional area, data analysis and reaction, we separately discuss data analysis (Table 5) and reaction (Table 6). Relevant features for data analysis extracted from the survey are:

- Default metrics and games analysis.** All game analytic platforms support a set of predefined metrics, algorithms and analysis tools that are available to process collected data. Predefined metrics are re-usable and possibly customizable metrics that address commonly measurable game aspects and are to be used out of the box. Most (commercial) platforms focus on monetary aspects (e.g., virtual money spent per session, amount of purchases), yet some also offer other default metrics (e.g., related to user engagement, mean time spent by a user in the game per day, social aspects such as invites and shares, or interaction with interface elements). Next to default metrics, all platforms also provide algorithms and analysis tools that can be used to analyze captured data, and/or to build more complex (custom) metrics with (see next bullet). For example, descriptive statistical methods are commonly offered (measures of central tendency, e.g., average, median; measures of variability, e.g., standard deviation, etc.); as they are common, we do not shown these in Table 5. Further algorithms and analysis tools include funnels (i.e., a series of steps towards a certain goal, a kind of behavior analysis), predictive analysis,

events correlation, frequency analysis, segments analysis (The user base can be segmented to target the analysis on such segments, or make metrics comparison between different segments (i.e., paid users, and trial users)), A-B test (A method for comparing two versions of the game, to determine which one performs better regarding some criteria), OLAP Cubes (A technique for analyzing multi-dimensional data), and Funnels (A technique for describing the navigation path followed by users in a system). Note that visual analytics are considered within the next functional area (Table 7).

- **Custom metrics.** Custom metrics allow developers to capture game-specific analytics, or extend already available metrics. A custom metric is hereby understood as a custom-defined distance function with an arbitrary level of complexity. Developers have a certain specification mechanism to their disposal (e.g., a restricted programming language), as well as the default game analytics tools and methods (see previous bullet) available to define their custom metric, and may extend default or other custom metrics. It is similar to coding user functions in any programming language. Custom metrics may be using input data captured as default event types, yet often they are based on custom event types as the underlying data model, required to model or structure a particular phenomenon. As such, the definition of custom metrics is linked to the ability to define custom event types. The possibility to specify new event types and custom metrics, beyond of default ones, is obviously a desirable characteristic for an analytics platform.
- **Spatial analysis.** As mentioned when discussing custom event types, we consider geospatial data support, and spatial analysis support as cross-cutting layers. With respect to metrics, this refers to regular metrics supporting geospatial data types, and specific spatial analytical technique or method supported or integrated in the analytics platform. As Table 5 shows, native support for spatial analysis is practically inexistent.

Table 5. Comparison of data processing and analysis related features among academic and commercial tools for game analytics. NA/NS = Not Applicable, Not Specified or Unknown.

| Tool/Citation | Default Metrics and Game Analysis | Custom Metric | Spatial Analysis |
|--------------------|---|---------------|------------------|
| WebTics [40] | Simple events correlation, frequency analysis | NA/NS | NA/NS |
| TRUE [41] | Video referencing | NA/NS | NA/NS |
| DataCracker [34] | NA/NS | NA/NS | NA/NS |
| Skynet [42] | Yes | NA/NS | Yes |
| GameAnalytics [43] | A/B-Tests, Funnels | NA/NS | NA/NS |
| HoneyTracks [44] | Custom segments, comparison, A/B-Tests | Yes | NA/NS |
| Xsolla [45] | Aggregations, etc. | NA/NS | NA/NS |
| GameGuts [46] | Aggregations, custom | Yes | NA/NS |
| DeltaDNA [47] | Funnels, Custom segments predictions, etc. | NA/NS | NA/NS |

Once metrics or analytical functions are computed, the outputs are usable in real-time, where game developers may program in-game reactions according to observed behavior (e.g., of players), or a posteriori, where output data from analysis is used for inspection, visualization or reporting. We treat the latter in more detail in the next functional area, and focus here on the reaction dimension.

Reactions from the game system often take the form of programmed rules that usually define what actions to take (storage, notification, altering game parameters or mechanics, etc.) when certain conditions, which often depend on the metric outputs, are met. As such, reactive behaviour can be regarded as reactive rules, either programmed or declarative, which are composed of conditions and actions. The ultimate aim for a game developer or analysts is to improve gameplay by provoking and/or incentivizing player behavior change through timely feedback/actions to players. Related features for reactive behaviour are depicted in Table 6:

- **Reactive rules.** When certain logical conditions are met over the results of data analysis, more specifically metric outputs, the system has the ability to react by triggering purposeful actions.

Typical actions with a specific aim are data storage (where possibly only a subset of the outputs is stored; supported by all surveyed platforms and thus not mentioned in Table 6) or notifications (e.g., to game developers or players). Further actions are often required to realize detailed, in-game consequences of observed behavior, for example to perform additional data transformation over the outputs of the metric or to trigger additional analytical tasks in cascade. In such a case, the ability for a game developer to *program* custom reactive actions is essential to optimize the game, and possibly trigger game mechanics updates. Obviously, actual alterations to the game are within the realm of the game implementation, not the analytics platforms. If a platform supports some form of reactive rules (i.e., logical conditions and consequent actions), we specify the type of action (storage, notification, or custom) in Table 6.

- **Geospatial support.** Output data from metrics may contain geospatial data structures, such as trajectories or polygons (e.g., representing the geographical area covered by a player). Such geospatial data may be subject to meet certain spatial or spatio-temporal conditions (e.g., within, intersect, crosses) to check, for example, whether a player's traveled distance during a game session is greater than 1 km, or the percentage of time a player remained inside a delimited area (often called "fence") is above average. Similarly, triggered actions can also be spatially-enabled. For example, actions that redistribute monitoring points or recalculate (a metric's) geofencing areas. For convenience, we split this feature into two columns in Table 6: geospatial support for conditions and actions, respectively.

Table 6. Comparison of reaction related features among academic and commercial tools for game analytics. NA/NS = Not Applicable, Not Specified or Unknown.

| Tool/Citation | Reactive Rules | Geospatial Support (Conditions) | Geospatial Support (Actions) |
|--------------------|----------------|------------------------------------|---------------------------------|
| WebTics [40] | Notification | NA/NS | NA/NS |
| TRUE [41] | NA/NS | NA/NS | NA/NS |
| DataCracker [34] | NA/NS | NA/NS | NA/NS |
| Skynet [42] | NA/NS | NA/NS | NA/NS |
| GameAnalytics [43] | NA/NS | NA/NS | NA/NS |
| HoneyTracks [44] | NA/NS | NA/NS | NA/NS |
| Xsolla [45] | NA/NS | NA/NS | NA/NS |
| GameGuts [46] | Custom | NA/NS | NA/NS |
| DeltaDNA [47] | NA/NS | NA/NS | NA/NS |

The fourth and last functional area comprises data visualization and reporting, and includes the following features (Table 7):

- **Data access.** Open access to data is useful to allow users to get access and query observed data, metrics results, or both, for further analysis by third-party tools. While this allows to harness the (extended) analytical power of external tool, in case the capabilities of the game analytics platform prove to be too general and/or insufficient, it also hinders integrability for real-time analysis and reactive rules generation. Indeed, access to analytical/output data is usually done for later off-line analysis. We broadly contemplate two strategies for data access: the first is based on on-line interfaces for querying analytical data (e.g., service end-points), and the other to enable a full export or bulk download.
- **Visual analytics.** It differs from games analysis in that the latter is thought to be part of metric computation to process collected/observed data. Visual analytical tools are situated at the end of the analytical pipeline (Figure 1), taking metric outputs as input (possibly after data transformations) to carry out in-depth analysis. Visual analytics comprises a large and varied arsenal of data visualisation and inspection techniques [48,49], in order to gain new insights and discover new patterns for game developers, which would otherwise be difficult to detect and/or

go unnoticed. Most of the surveyed tools and platforms only support basic techniques, such as different types of charts to visually summarise and aggregate data.

- **Geospatial support.** Similar to the geospatial support for game analysis presented earlier, here we refer to specific spatial and spatio-temporal support for visual analytics, commonly referred to in literature as visual geanalytics [50]. Examples include visualization for moving point datasets (e.g., players position in multiplayer games) or time series (e.g., virtual land ownership or visibility over time). This area of research is quite extensive and developed methods and techniques have been proven to be effective for decision making [51]. Only a few analytical tools and platforms (e.g., [42]) offer basic support, in the form of map visualisation of simple data types like points or heatmaps (density maps of spatial frequency).

Table 7. Comparison of data reporting and visualization related features among academic and commercial tools for game analytics. NA/NS=Not Applicable, Not Specified or Unknown.

| Tool/Citation | (Output) Data Access | Visual Analytics | Geospatial Support |
|--------------------|----------------------|---------------------------------------|----------------------|
| WebTics [40] | NA/NS | NA/NS | NA/NS |
| TRUE [41] | NA/NS | Reports | NA/NS |
| DataCracker [34] | NA/NS | Charts, filtered reports | NA/NS |
| Skynet [42] | NA/NS | Charts, filtered reports | Point maps, heatmaps |
| GameAnalytics [43] | Yes | Charts, filters, comparison | NA/NS |
| HoneyTracks [44] | Yes | Charts, reports, cohorts, comparison | NA/NS |
| Xsolla [45] | Yes | Charts, reports | NA/NS |
| GameGuts [46] | NA/NS | NA/NS | NA/NS |
| DeltaDNA [47] | Yes | Charts, reports, funnels, predictions | NA/NS |

3.3. Limitations

Looking at Tables 3–7 we identify some limitations of the reviewed analytical tools and platforms with respect to their geospatial support, and hence, to the creation, computation and visualisation of metrics for location-aware games.

A first general observation is that virtually all columns that refer to “geospatial support” are empty, i.e., geospatial characteristics and techniques are poorly supported. This is a direct result of the fact that none of the surveyed tools and platforms were specifically designed for defining and computing spatio-temporal metrics. Nevertheless, while built-in support is largely missing, some platforms allow to treat simple geospatial concepts, such as data models to capture and store location and points. In some platforms (Table 4), the definition of a data model for capturing location (e.g., player position) is straightforwardly possible, since a position is a rather simple data structure (i.e., event type) composed of a pair of coordinates. In most cases, default event types suffice as a point data model. Other types of more complex geometries, such as lines, poly-lines, or polygons are not mentioned or not supported. This suggests that the most popular game analytical platforms are merely restricted to natively supporting point geometries.

Nevertheless, the ability to structure and manage a player’s position does not necessarily imply that this bit of spatial information is properly exploited and analyzed. Beyond of the simple visualization of player position onto a map (Table 7), which represents a simple analysis method over point geometries, more advance spatial analysis techniques like topological operators, are absent. This deficiency—the lack of a geospatial toolbox and the support for complex geometries beyond point data models—considerably limits the use of the surveyed analytical platform for defining advanced custom metrics for location-aware games, even though (non-spatial) games analysis and metrics are generally well covered (Table 5). We therefore argue that, to the best of our knowledge, spatio-temporal analytical methods for metric computation are not the focus, and thus not properly supported, in the current game analytical tools. Simply put, spatial analysis is passed over in favor of mainstream analytics methods for traditional metric scenarios in games (Table 5).

Another observation is whether there exists a relation between data collection strategies and the subsequent type of game analysis supported. For example, if a platform supports streamed events as an strategy for data communication (Table 3), does it mean that stream computing (e.g., [52]) is also covered? Does the platform meet the demand for fast monitoring and storage of huge amounts of data in real-time? Unfortunately, we do not observe a clear pattern with respect to this matter. Indeed, those platforms that claim to support streamed events do not exhibit any type of real-time data analytics. Regardless of the data communication strategies (data streaming, data upload), the type of game analytics supported is quite homogeneous over the tools but constrained to well-known, off-line, and non-spatial techniques for metric analytics (see [19]), especially for monitoring and measuring any variable related to the actual behavior of a player (gameplay). We may consequently argue that real-time analytics and stream processing is a concern that requires support throughout all functional areas.

We have just stated that the type of game analytics supported is quite homogeneous, in the sense that most tools and platforms offer pretty much the same core set of game analysis techniques (see Table 5). The other side of the coin is the support for user-defined or custom metrics. Here, the degree of personalisation is quite marked, resulting in two opposed groups of tools. One group ([34,40–42]) does not consider customisation at all, and thereby, custom metrics and associated data representations are not configurable (or the process of customisation introduces a significant complexity and additional effort from the game developer). The other group ([43–47]) supports both custom event types and custom metrics, which makes sense for the specification of user-defined metrics. Obviously, as we mentioned earlier, custom metrics as the way to specify new event types and custom functions (metrics), beyond of default ones, is a desirable characteristic for game analytics platforms.

In conclusion, what is missing for supporting metrics oriented to location-aware games? Responding to this question is challenging since there is no native analytical platform for location-aware games. Paradoxically, the ever-increasing importance of the where and location in varied game genres (e.g urban games, adventure,) situated in real places, is not being reflected with the same intensity in the research and development of analytical platforms for spatio-temporal metrics computation. Current trends suggest that both commercial and academic game analytical tools are shyly glancing at the geospatial field, but there remains a lot to be done, especially to cover (advanced) spatial analysis throughout all functional areas. In the next section, we attempt to identify what spatial techniques are required, by proposing a categorisation of spatial metrics, in order to facilitate, promote and push for game analytical platforms (gradually) supporting various types of spatio-temporal metric computation.

4. Classification of Metrics for Location-Aware Games

From the argumentation of the previous section we can summarise that location-aware games and games in general share many aspects, specifically what concerns measuring gameplay. This makes perfect sense as the improvement of gameplay is a recurrent objective for game developers and analysts regardless of the nature of game. However, we also remarked important differences between the two, especially due to the fact that spatial analysis deals with the real-life dimensions of play, i.e, the actual location, environment and geography are defining characteristics of the mechanics of location-aware games. Logically, spatial analytics techniques are required to compute and monitor spatial-related aspects of the game. Nevertheless, our analysis showed that this is precisely the missing feature in the game analytics platforms and tools considered in Section 3.

In this section, we propose a classification of spatially-related metrics for location-aware games and position it with respect to other existing classifications of non-spatial metrics. By doing so, we pursue two main objectives:

1. to make spatial metrics visible by placing them at the same level and embedding them within the larger context of metrics for games. Higher visibility may lead to the next generation of analytics

platforms natively supporting spatial data and spatially-related metrics, increasingly recognising the strategic importance of managing location and spatial concepts in games.

2. to bring the gaming and GIScience research communities' attention to location-aware games as an interesting and relatively unexplored research field, with ample possibility for additional research and practical tool development that may directly impact applications in a wide variety of application fields, such as (smart) cities (e.g., urban games), health (e.g., games to stimulate physical activity) or sociology (e.g., games to stimulate social inclusion).

4.1. Existing Metrics Classifications

As emphasised in Section 2, the focus of the paper is on metrics related to gameplay, i.e., to evaluate game design, user experience and user behaviour. Gameplay considers the player as the center of the metrics process. Drachen et al. [53] exemplified this vision in their high-level classification for game metrics (top part of Figure 2). Looking at the Gameplay box, the authors classified metrics for gameplay into three main groups: Interface, In-game and System. Interface metrics are about interactions of the player with the game interface (e.g., interactions with menus and buttons, mouse sensitivity); In-game metrics cover players' actions and behaviors during the game (e.g., players' position, their interactions with game elements such as objects or resources); and System metrics cover the actions game systems initiate to respond to player actions (e.g., refresh game resources once a player reaches a pre-defined set of conditions).

In the literature, there exist various proposals for game metrics classification mainly related to Interface and In-game groups, as these two accrue most of the existing examples of game metrics. For example, Tychsen and Canossa [31] proposed four specific game metrics, namely navigation, interaction and narrative metrics, which fall into the In-game group, and interfaces metrics, which are quite similar to the Interface group. Besides, the authors strongly linked these types of metrics to data collection strategies. Navigation metrics often require data collection via time-based strategies (at regular rates), while interaction metrics are well suited for event-based data collection strategies. Nacke et al. [54] took a more social perspective of In-game metrics and proposed specific types of game metrics such as social metrics (e.g., number of friends, of challenges, etc.) and viral metrics (e.g., number of invitations, shares, kudos, etc.), among others. The focus was to measure social aspects and behaviors of games, which were often based on the Facebook platform, thereby capturing the strong social dimension of this type of games. Bernhaupt and Mueller [55] classified metrics depending on genre/type of game they are used in. They distinguish between generic gameplay metrics (for features that are common in all games, e.g., game session time, total playing time), genre-specific gameplay metrics (for games in a same category that share common features, e.g., in shooter-based games: time spent using a certain weapon), and game-specific gameplay metrics (metrics that are based on game-specific unique features and that are custom designed, e.g., usage of special abilities of a character). This classification can be considered orthogonal to Drachen et al. [53]'s classification, as it provides an alternative view, according to context of use (type/genre of game) rather than functional context (Interface, In-game, System), on the metrics landscape.

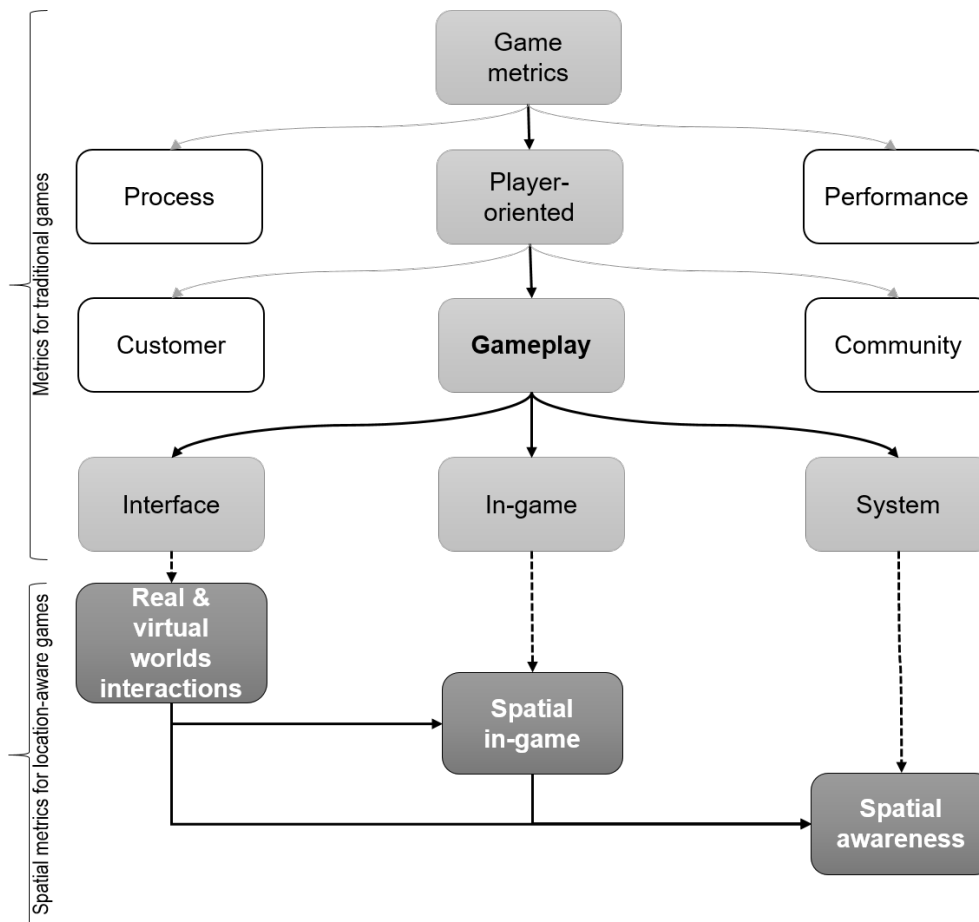


Figure 2. Spatial metrics classification embedded into existing classification of game metrics centered on gameplay/player (extended from [53]).

4.2. Spatial Metrics Classification

Rather than proposing a new classification, we consider Drachen et al. [53]’s high-level, schematic classification as an appropriate starting point to embed spatial metrics within the realm of traditional metrics for gameplay. We hereby emphasise that both types of metrics can and *should* work in cooperation, and we recognize the importance and impact the functional context (interface, in-game, system) of metrics may have in practice (e.g., for game analytics platforms implementations). As such, we extend Drachen et al. [53]’s classification to include the spatial notions critical for location-aware games, see bottom part of Figure 2. Nevertheless, we note that the inclusion of the spatial context of location-aware games, and the spatial metrics that address them, requires a slight re-interpretation of the three metric groups (Interface, In-game, System).

First, we note that the In-game and Interface categories need to be more broadly interpreted. Furthermore, both are more tightly coupled and influence each other. Indeed, in location-aware games, the real world may be (at least partly) embedded in the game, whereby it influences or becomes the playing field. In other words, the real world becomes part of the In-game context. Therefore, In-game metrics need to address “regular” In-game features, yet also include spatial metrics aimed at monitoring behaviour of a player in the real world, which is possibly influenced by the surrounding environment. Furthermore, when real and virtual worlds blend, metrics mixing virtual features with real-world (spatial) features are necessary.

On the other hand, interactions with the game become tightly coupled with the real world, as e.g., game resources (avatars, enemies, etc.) may be embedded both in the virtual and the real world. For the latter case, they are strongly dependent on the (spatial) characteristics of the real world.

Therefore, while metrics for Interface and In-game were treated in relative isolation in traditional games, this is no longer exclusively the case in location-aware games. Notwithstanding this tighter coupling between Interface and In-game contexts, both also still exhibit distinctive features, which may be studied independently. For example, game developers may utilise location-based technology (e.g., augmented reality, context-aware mobile interfaces), and as such, how players interact with the user interface of the game application is still important and subject to monitoring within the realm of Interface metrics.

The resulting classification in Figure 2 shows the newly added metrics groups, addressing geospatial features of location-aware games, along with their relation to each other and to the original groups. The geospatially related groups refer to what the target of the analysis of the metrics is, namely “Real & virtual world interaction”, “Spatial In-game” and “Spatial awareness”. Arrows among them denote the interaction between these metrics, which can not always be seen in isolation. For example, player’s actions (Real & virtual worlds interaction) in location-aware games leave spatial footprints that can help to better explain player behaviour in the game (Spatial in-game) than when that latter aspect is studied without spatial context.

We hereby note that, from a GIScience point of view, the different types of metrics defined in the well-known classification of de Smith et al. [56] based traditionally on univariate, bivariate, and multivariate spatial analysis, are applicable. Choosing the right type of spatial analysis, based on the number of spatial variables involved, is up to game developers/analysts and the scope of the metric. For example, trajectory-related metrics can measure movement of a player, or a group of players (clustering), or be used in combination with event data (number of deaths, picked up resources, etc.). In these cases, the process of computing trajectories may involve one, two or more variables and thus require different spatial techniques (see [56]), even though all these examples belong to the same logical group of metrics (Spatial in-game). In other words, our metrics are grouped based on functional context, just as in Drachen et al. [53]’s original classification, rather than on the technicality of the used analysis technique, as in [56].

4.3. Spatial Metrics

Spatial event types, i.e., (logical) data models that capture geospatial concepts, along with geospatial processing algorithms and analysis tools, are essential to support spatial metrics. Spatial metrics are particularly important for the Spatial in-game group, where they use spatial properties of the player collected data to compute and reason about the spatial behavior of players. Nevertheless, as discussed throughout Section 3, support for geospatial event types and processing is equally relevant for the Interface and System metrics groups, where they allow to take geospatial properties of interface- (e.g., *where* do game interactions take place?) and system-related (e.g., *where* are system-related actions taking place?) actions into account. In the remainder of this section, we discuss different types of spatial metrics according to the spatial properties they utilize (summary in Table 8), and primarily relate them to the most important metrics group, Spatial in-game metrics.

- **Location-based metrics (point-based metrics).** This category includes metrics that take into account specific locations (i.e., most often represented by *coordinates*), for example, the location of a player or resource. Point-based metrics are the simplest, yet they may reveal interesting properties of the player, interface and game as a whole. For example, point-based metrics may determine stationary behavior (e.g., an “ambush” in a shooter-based game); proximity or line of sight among and between players, resources, and environmental features; and altitude difference or altitude-related computations. Point-based metrics may reveal relevant game properties, such as significant locations (e.g., locations that attract players) or imbalances (e.g., locations which provide an (unfair) advantage or disadvantage). Example papers supporting this type of metrics include Coulton et al. [29], where location-based metrics are employed to analyze players behavior in concrete location-based games. In particular, the distance between players as they moved around the game area, and also the mean distance from the average position, is used as

- a measure of “how adventurous each player was during the game”. Some other examples are presented in [57], in that location is used for analyzing where deaths occur in the game, or where the users request help, for determining those places where possible imbalances might be present. In [32] the speed is used for determining spots in the game where the player was less challenged.
- **Trajectory metrics.** This category includes metrics that take the trajectory of the players (or resources) into consideration, for example, the distance between the trajectories of two players; the time spent in a trajectory; rhumb; the convergence or divergence of trajectories; trajectory patterns; etc. can be candidates for metrics, or elements to consider in metrics. Example papers supporting this type of metric include Hoobler et al. [58], where the player trajectory analysis can provide information about the players or team strategies, as well as resources (e.g., trajectories of fires/shots). In [39] the importance of trajectory analysis is highlighted in order to detect bots, analyzing different types of user behaviors (cooperation, flock etc), and detection of user strategies that can be indicators of game imbalance problems. Coulton et al. [29] also used trajectories as a basis for analysis of players behavior (and unusual behaviors) during the game.
 - **Space-related metrics (area-related metrics).** This category includes metrics which perform calculations based on areas, for example, in the centre, forest, sea; overlapping/disjoint/contained areas (e.g., action radius); inclusion/exclusion of a player/resource in an area (i.e., geofencing); explored/covered area; exploration speed; area exploitation; spreadness/distribution of resources (i.e., rewards); and so on. Example papers supporting this type of metrics include Hoobler et al. [58], who suggested clustering techniques for evaluating the distribution of game resources or for design validation (“planned for affordance”). This is especially important for location-aware games, where access to resources usually involves a physical effort. In [59] clustering techniques are used for identifying areas where the users behave more fiercely in the game. Similarly, Wallner and Kriglstein [60] used clustering techniques for providing insights about the distribution of changes in states of the game. In [32] the analysis is based on comparing player’s behavior in selected areas of the game. Finally, geofencing techniques are exemplified by Map Attack ESRI [61], while Martí et al. [62] discussed its application for noise pollution monitoring.

Table 8. Summary of examples of spatial metrics.

| Type | Examples | Example Papers |
|--------------|--|--|
| Point-based | <i>Distance:</i> between players’ position; mean distance from the average position | Coulton et al. [29] |
| | <i>Location:</i> where (point cloud) deaths occur, players request help; where players are less challenged | Drachen and Canossa [32], Drachen and Canossa [57], |
| Trajectory | <i>Analysis:</i> distance, convergence, etc. of either players’ or resources’ | Hoobler et al. [58] |
| | <i>Patterns:</i> trajectory patterns for user behaviour | Drachen and Schubert [39], Coulton et al. [29] |
| Area-related | <i>Clustering:</i> clustering techniques for resources/players distribution | Hoobler et al. [58], Wallner and Kriglstein [59] |
| | <i>Geofencing:</i> inclusion/exclusion of players/resources in a area (fence); virtual urban zoning | ESRI [61], Martí et al. [62] |

5. Discussion

5.1. Spatial Metrics to Improve Gameplay in Location-Aware Games

As it can be expected and was reasoned throughout this article, many of the metrics used in regular games can also be used in location-aware games. Indeed, it can be argued that general metrics in games are a subset of location-aware games metrics. For example, metrics related to user engagement (daily game usage, actions performed, levels played, time spent per session) or monetary

metrics (statistics about money spent in digital items) can be used in location-aware games, as they are general enough to be used in most kind of games. Nevertheless, the particularities of location-aware games demand the inclusion of spatial properties, both for existing metrics (e.g., for monetary metrics, statistics about money spent at a specific location) and for specific (spatial) metrics (e.g., for game balance, the spatial distribution of player actions/events, such as deaths, time spent.).

Given the nature of the playground in location-aware games (real, physical world), the changing conditions, and the characteristics of the terrain (i.e., type of terrain, altitude, slope) can influence gameplay. Spatial metrics (see Section 4.3) come into play to measure these characteristics, and encompass the three previously mentioned types of spatial metrics. Although the conditions might be the same for all players, in cases where the game can be played in different places or times, the playground conditions can be determinant with respect to the players' performance and/or be a source of imbalance, e.g., assessing the "effort" needed for reaching the resources (given the physical component that may be involved with location-aware games), determining valid routes, etc. Terrain type, altitude, weather conditions and so on can be monitored by spatial metrics to extract useful information, such as mean slope of the terrain, light conditions (day, night), terrain visibility (e.g., open field, a city, a mountain, forest), humidity, temperature, wind speed, altitude etc. For example, Herold et al. [63] evaluated the role of spatial metrics in the context of urban land use change modeling. The spatial metrics are defined as "measurements derived from thematic-categorical maps, exhibiting spatial heterogeneity at a specific scale, and resolution."

Player's characteristics such as physical attributes and conditions [28] (i.e., weight, locomotion characteristics), and geography awareness/knowledge of the place where the game is taking place, can also be considered as contributing factors for the "effort or difficulty" the player has to employ to participate in the game, and regardless of other types of games, are aspects to consider when evaluating the playability and the engagement of the users.

The spatial awareness between players and the actual spatial dimension and configuration of the game is significant in location-aware games. Again, this encompasses the three previously mentioned spatial metric types, and quantifies the relation of the player with the geographical space, such as interaction with the map (map visibility, capacity for exploring/obtaining information about the map), previous knowledge of the map/place, availability of the location and the location device. For example Kremer et al. [64] empirically discussed how the spatial decisions of a player (i.e., deciding which point of interest to visit next, or where to start searching) have an impact in the overall learning experience.

Finally, on the downside, there exist difficulties and open issues in designing location-aware games. Jacob and Coelho [28] provide a concise account of these difficulties: adapting the game to the player's location; lack of availability of location-based information; player's fitness to game physical requirements; hardware limitations; and data protection and privacy concerns. Besides, we add software challenges for materialising location-aware games. With an increased amount of (spatial) data to consider, location-aware games face the challenge of managing, processing and taking into account such data in real time. While massive on-line multi-player games have successfully coped with the challenge of handling data of millions of players, the increased strain of the additionally monitoring, processing and taking into account geospatial data provides a significant software challenge, which yet needs to be addressed.

5.2. Implication for Future Geospatial Game Analytics Platforms

This research is part of a larger research roadmap that aims to provide a game analytics platform that supports spatial metrics for location-aware games. As a first step, in this article, we explored the literature for existing support, and categorized and characterized spatial metrics. As further steps, we plan to develop a conceptual platform that addresses all required features to measure spatio-temporal aspects of location-aware games, as uncovered in this article, and develop the actual

platform. As such, let's shortly discuss the implications of the analysis and findings presented in this article on the (architecture and implementation) of this future geospatial analytics platform.

As a first observation, scalability is an important factor to take into account, as location-aware games may attract a potentially huge number of players (e.g., Pockemon Go had thousands of daily active users [8]), with huge amounts of data gathered. Evidently, the same is true for other classes of games, such as (regular) massively multiplayer online games (MMOGs), yet the computational complexity significantly increases for location-aware games as geospatial data and computations come into play. In addition, the desired reactive capabilities of the analytics platform, (e.g., perform notifications based on conditions applied to the results of spatial metrics computations), puts an additional computational strain on future geospatial analytics platform. All of this has direct implications on the architecture of the system, which necessarily needs to deploy techniques for big data, computational decentralization and real-time processing (e.g., data distribution and replication, distributed computing, etc.) with associated technologies (e.g., distributed NoSQL databases, such as Hadoop, cluster computing frameworks, such as Apache Spark).

Secondly, a rich built-in support for geospatial data representation and event types, beyond simple point-based coordinates (e.g., spatio-temporal trajectories, polygon, 3D objects, etc.), with associated geospatial computation methods, is required. Offering such native support allows performance gains due to internal optimizations in storing and handling geospatial data, and easier use and exploitation of geospatial features (by developers) through pre-defined geospatial functions and metrics. A wealth of work in this respect is available in the Geographical Information Science (GIScience) community (e.g., data storage optimizations, spatial algorithms), which can be re-used in the future geospatial metrics framework. Furthermore, compatibility with existing standards may facilitate interfacing with and use of existing open source and commercial software specialized in geospatial data storage and computations.

Thirdly, in extension to the previous point, extensibility is essential to support a broad variety of location-aware scenarios (including those using potentially new or upcoming technologies, such as augmented reality). While predefined geospatial event types are important (e.g., for optimization purposes), a lack of extensibility limits the flexibility of the system to handle unforeseen or changing requirements. The envisioned platform should be flexible for modelling any type of data and ideally provide convenient mechanisms to handle and propagate it to other platform's components so that such information about data model can be purposed for metric definition, computation, and validation. This implies support for custom metrics definition, both native and custom event types and geospatial computations. This level of integration necessarily requires a flexible architecture, which allows data models and metric definitions to be accessible all over the platform (i.e., both at client and server side). While this level of flexibility might carry some negative implications (e.g., storage of arbitrary complex data models may impede performance), the benefits would overpass by far the limitations incurred by limited extensibility, as generally found in the current state-of-the-art.

Finally, and hand in hand with custom event types as discussed in the previous point, the future geospatial analytics platform should support custom geospatial computation and custom metrics specification and execution at the server side, accompanied by convenient tools and APIs to support developers in handling the full metrics life-cycle (i.e., definition, execution, spatial analysis, interpretation of results). In designing such tools and APIs, openness of data and analytics services hereby need to be balanced with data privacy, as these may be suspect to location-based data mining attempts/attacks (e.g., see [65,66]).

6. Conclusions and Future Work

Metrics measure player's behavior and game's properties. Several research works and commercial platforms exist that allow game developers to address the different aspects involved in metrics deployment: data specification and collection, metrics definition and calculation, and analysis and visualization. In this article, we focus specifically on metrics for location-aware

games. To unambiguously set the context, we first discuss definitions of location-aware games, metrics, and related terms. We then present a literature review of support for spatio-temporal features in existing game analytics platforms, based on a conceptual characterization of the latter. We distinguished four functional areas: data collection and communication, data representation, data analysis and reaction, and data visualization and reporting. For each area, we discussed the main features and their support in existing analytics platforms in detail, hereby paying special attention to spatial characteristics as required by location-aware games. Our analysis reveals a clear lack of support for spatial metrics. In a first step to full recognition and support for spatial metrics, we subsequently further analyse spatial metrics in the realm of game metrics, and propose a classification for spatial metrics integrated in existing classes of game metrics: real and virtual world interactions (related to user interface, regarding both the virtual and real world interactions), spatial in-game (related to player's actions and behavior in the game, both in the virtual and real world) and spatial awareness (regarding spatially-related decisions and action the game initiates). With this extended classification, we suggest the importance of geospatial features throughout all aspects of location-aware games. Indeed, we argue that general game metrics are equally applicable in location-aware games, yet the specific nature of location-aware games adds spatial and temporal dimensions which may warrant extending traditional metrics (e.g., tracking location of monetary transactions or defining new ones (e.g., difficulty with respect to physical terrain features). In this setting, we further elaborate three types of spatial metrics, namely point-, trajectory- and area-based metrics, mainly relevant for the in-game category, provide examples and discuss difficulties. As for the latter, in addition to general difficulties in implementing location-aware games as reported by Jacob and Coelho [28] (taking into account players' location, availability of location-based data, players' fitness, hardware limitations, data protection and privacy concerns), we identify the additional difficulty of lack of suitable (scalable) software support for the geospatial features and computation (including geospatial metrics platforms).

Finally, we discuss the implications of the presented analysis of existing game analytics platforms towards future platforms that support the geospatial features typically required by location-aware games. The main observations and implications are: (i) scalability, both with respect to data and (real-time) computational requirements, suggesting support for big data handling, computational decentralization and real-time processing; (ii) need of built-in data representation and event type support, with associated compatibility with existing methods and tools; (iii) platform extensibility, requiring an open and flexible platform architecture; and (iv) custom metrics support and supporting APIs and analysis tools, balancing analytical strength with privacy concerns.

We believe that such a platform that allows the flexible definition and deployment of native and extendable, configurable spatial metrics opens up opportunities for location-aware games in various application areas. We plan to explore how such an analytics platform for location-aware games may be beneficial in developing novel games in the fields of education (e.g., games pursuing specific learning goals), health (e.g., games pursuing health objectives, such as increased physical activity) and smart cities (e.g., games improving citizen engagement).

Acknowledgments: Sven Casteleyn and Carlos Granell are funded by the Ramón y Cajal Programme of the Spanish government (grant numbers RYC-2014-16606 and RYC-2014-16913, respectively).

Author Contributions: Luis E. Rodríguez-Pupo conducted the comparative analysis. Luis E. Rodríguez-Pupo, Sven Casteleyn, and Carlos Granell wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Avedon, E.M.; Sutton-Smith, B. *The Study of Games*; John Wiley: New York, NY, USA, 1971.
2. Salen, K.; Zimmerman, E. *Rules of Play; Game Design Fundamentals*; MIT Press: Cambridge, MA, USA, 2004.
3. Ritterfeld, U.; Cody, M.; Vorderer, P. *Serious Games: Mechanisms and Effects*; Routledge: Abingdon, UK, 2009.

4. Johnson, D.; Horton, E.; Mulcahy, R.; Foth, M. Gamification and serious games within the domain of domestic energy consumption: A systematic review. *Renew. Sustain. Energy Rev.* **2017**, *73*, 249–264.
5. Pedreira, O.; García, F.; Brisaboa, N.; Piattini, M. Gamification in software engineering—A systematic mapping. *Inf. Softw. Technol.* **2015**, *57*, 157–168.
6. Duggan, E. Squaring the (Magic) Circle: A Brief Definition and History of Pervasive Games. In *Playable Cities: The City as a Digital Playground*; Nijholt, A., Ed.; Springer: Singapore, 2017; pp. 111–135.
7. Schlatter, B.E.; Hurd, A.R. Geocaching: 21st-century hide-and-seek. *J. Phys. Educ. Recreat. Dance* **2005**, *76*, 28–32.
8. Smith, C. 80 Amazing Pokemon Go Statistics and Facts (September 2017). Available online: <https://expandedramblings.com/index.php/pokemon-go-statistics/> (accessed on 26 September 2017).
9. Schlieder, C.; Kiefer, P.; Matyas, S. Geogames: Designing location-based games from classic board games. *IEEE Intell. Syst.* **2006**, *21*, 40–46.
10. Schlieder, C.; Kiefer, P.; Matyas, S. Geogames: A conceptual framework and tool for the design of location-based games from classic board games. In Proceedings of the International Conference on Intelligent Technologies for Interactive Entertainment, Madonna di Campiglio, Italy, 30 November–2 December 2005; Springer: Berlin, Germany; pp. 164–173.
11. González Sánchez, J.L.; Padilla Zea, N.; Gutiérrez, F.L. From Usability to Playability: Introduction to Player-Centred Video Game Development Process. In Proceedings of the 1st International Conference on Human Centered Design, San Diego, CA, USA, 19–24 July 2009; Springer: Berlin, Germany; pp. 65–74.
12. Jaffe, A.; Miller, A.; Andersen, E.; Liu, Y.E.; Karlin, A.; Popovic, Z. Evaluating Competitive Game Balance with Restricted Play. In Proceedings of the Eighth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, Stanford, CA, USA, 8–12 October 2012; pp. 26–31.
13. Kiefer, P.; Matyas, S. The Geogames Tool: Balancing spatio-temporal design parameters in location-based games. In Proceedings of the 7th International Conference on Computer Games, Angoulême, France, 28–30 November 2005; pp. 216–222.
14. Reis, J.P.; Silva, E.A.; Pinho, P. Spatial metrics to study urban patterns in growing and shrinking cities. *Urban Geogr.* **2016**, *37*, 246–271.
15. García, F.; Pedreira, O.; Piattini, M.; Cerdeira-Pena, A.; Penabad, M. A Framework for Gamification in Software Engineering. *J. Syst. Softw.* **2017**, *132*, 21–40.
16. Bhatti, S.S.; Reis, J.P.; Silva, E.A. Spatial Metrics: The Static and Dynamic Perspectives. In *Reference Module in Earth Systems and Environmental Sciences*; Elsevier: Amsterdam, The Netherlands, 2017.
17. Seaborn, K.; Fels, D.I. Gamification in theory and action: A survey. *Int. J. Hum.-Comput. Stud.* **2015**, *74*, 14–31.
18. Nacke, L.E.; Deterding, S. The maturing of gamification research. *Comput. Hum. Behav.* **2017**, *71*, 450–454.
19. El-Nasr, M.S.; Drachen, A.; Canossa, A. *Game Analytics: Maximizing the Value of Player Data*; Springer: London, UK, 2013.
20. Drachen, A.; Schubert, M. Spatial Game Analytics. In *Game Analytics: Maximizing the Value of Player Data*; El-Nasr, M.S., Drachen, A., Canossa, A., Eds.; Springer: London, UK, 2013; pp. 365–402.
21. Fürnkranz, J. Machine learning and game playing. In *Encyclopedia of Machine Learning*; Sammut, C., Webb, G.I., Eds.; Springer: Berlin, Germany, 2011; pp. 633–637.
22. Southey, F.; Xiao, G.; Holte, R.C.; TROMMELN, M.; Buchanan, J.W. Semi-Automated Gameplay Analysis by Machine Learning. In Proceedings of the First Artificial Intelligence and Interactive Digital Entertainment Conference, Marina del Rey, CA, USA, 1–3 June 2005; pp. 123–128.
23. Prensky, M. The motivation of gameplay: The real twenty-first century learning revolution. *On the Horizon* **2002**, *10*, 5–11.
24. Rollings, A.; Ernest, A. *Andrew Rollings and Ernest Adams on Game Design*; New Riders: Indianapolis, IN, USA, 2003.
25. Costkyan, G. I have no words & I must design: Toward a critical vocabulary for games. In Proceedings of the Computer Games and Digital Cultures Conference, Tampere, Finland, 6–8 June 2002.
26. Kiili, K. Digital game-based learning: Towards an experiential gaming model. *Internet High. Educ.* **2005**, *8*, 13–24.

27. Montello, D.R. Scale and multiple psychologies of space. In Proceedings of the European Conference on Spatial Information Theory, Marciana Marina, Italy, 19–22 September 1993; Springer: Berlin, Germany; pp. 312–321.
28. Jacob, J.T.P.N.; Coelho, A.F. Issues in the development of location-based games. *Int. J. Comput. Games Technol.* **2011**, *2011*, doi:10.1155/2011/495437.
29. Coulton, P.; Bamford, W.; Cheverst, K.; Rashid, O. 3D Space-time visualization of player behaviour in pervasive location-based games. *Int. J. Comput. Games Technol.* **2008**, *2008*.
30. Weisstein, E.W. Metric. MathWorld—A Wolfram Web Resource. Available online: <http://mathworld.wolfram.com/Metric.html> (accessed on 19 July 2017).
31. Tychsen, A.; Canossa, A. Defining personas in games using metrics. In Proceedings of the 2008 Conference on Future Play: Research, Play, Share, Toronto, ON, Canada, 3–5 November 2008; ACM: New York, NY, USA; pp. 73–80.
32. Drachen, A.; Canossa, A. Towards gameplay analysis via gameplay metrics. In Proceedings of the 13th International MindTrek Conference: Everyday Life in the Ubiquitous Era, Tampere, Finland, 30 September 2009; ACM: New York, NY, USA; pp. 202–209.
33. Hochleitner, C.; Hochleitner, W.; Graf, C.; Tscheligi, M. A Heuristic Framework for Evaluating User Experience in Games. In *Game User Experience Evaluation*; Bernhaupt, R., Ed.; Springer: Cham, Switzerland, 2015; pp. 187–206.
34. Medler, B.; John, M.; Lane, J. Data cracker: Developing a visual game analytic tool for analyzing online gameplay. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Vancouver, BC, Canada, 7–12 May 2011; ACM: New York, NY, USA; pp. 2365–2374.
35. Kaner, C.; Bond, W. Software engineering metrics: What do they measure and how do we know. In Proceedings of the 10th International Software Metrics Symposium, Chicago, IL, USA, 11–17 September 2004; pp. 1–12.
36. Stanton, M.; Humberston, B.; Kase, B.; O'Brien, J.F.; Fatahalian, K.; Treuille, A. Self-refining games using player analytics. *ACM Trans. Graph.* **2014**, *33*, 73.
37. Drachen, A.; Thureau, C.; Sifa, R.; Bauckhage, C. A comparison of methods for player clustering via behavioral telemetry. In Proceedings of the Foundations of Digital Games, Chania, Greece, 14–17 May 2013; pp. 245–252.
38. Jimenez, E.; Mitchell, K.; Seron, F. Capture and analysis of racing gameplay metrics. *IEEE Softw.* **2011**, *28*, 46.
39. Drachen, A.; Schubert, M. Spatial game analytics and visualization. In Proceedings of the IEEE Conference on Computational Intelligence in Games, Niagara Falls, ON, Canada, 11–13 August 2013; pp. 1–8.
40. McCallum, S.; Mackie, J. WebTics: A Web Based Telemetry and Metrics System for Small and Medium Games. In *Game Analytics: Maximizing the Value of Player Data*; Seif El-Nasr, M., Drachen, A., Canossa, A., Eds.; Springer: London, UK, 2013; pp. 169–193.
41. Kim, J.H.; Gunn, D.V.; Schuh, E.; Phillips, B.; Pagulayan, R.J.; Wixon, D. Tracking real-time user experience (TRUE): A comprehensive instrumentation solution for complex systems. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Florence, Italy, 5–10 April 2008; ACM: New York, NY, USA; pp. 443–452.
42. Zoeller, G. Game Development Telemetry in Production. In *Game Analytics: Maximizing the Value of Player Data*; El-Nasr, M.S., Drachen, A., Canossa, A., Eds.; Springer: London, UK, 2013; pp. 111–135.
43. GameAnalytics Web Site. Available online: <http://www.gameanalytics.com/> (accessed on 9 August 2017).
44. HoneyTracks Web Site. Available online: <https://honeytracks.com/> (accessed on 9 August 2017).
45. Xolla Analytics Web Site. Available online: <https://xsolla.com/modules/analytics/> (accessed on 9 August 2017).
46. Albuquerque, M.T.C.F.; Ramalho, G.L.; Corruble, V.; Santos, A.L.M.; Freitas, F. Helping Developers to Look Deeper inside Game Sessions. In Proceedings of the 2014 Brazilian Symposium on Computer Games and Digital Entertainment, Porto Alegre, Brazil, 12–14 November 2014; pp. 31–40.
47. DeltaDNA Web Site. Available online: <https://deltadna.com/> (accessed on 9 August 2017).
48. Dill, J.; Earnshaw, R.; Kasik, D.; Vince, J.; Salen, P.C.W.; Zimmerman, E. *Expanding the Frontiers of Visual Analytics and Visualization*, 1st ed.; Springer: London, UK, 2012.
49. Fry, B. *Visualising Data*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2008.

50. Andrienko, G.; Andrienko, N.; Demsar, U.; Dransch, D.; Dykes, J.; Fabrikant, S.I.; Jern, M.; Kraak, M.J.; Schumann, H.; Tominski, C. Space, time and visual analytics. *Int. J. Geogr. Inf. Sci.* **2010**, *24*, 1577–1600.
51. Andrienko, G.; Andrienko, N.; Jankowski, P.; Keim, D.; Kraak, M.J.; MacEachren, A.; Wrobel, S. Geovisual analytics for spatial decision support: Setting the research agenda. *Int. J. Geogr. Inf. Sci.* **2007**, *21*, 839–857.
52. Garofalakis, M.; Gehrke, J.; Rastogi, R. *Data Stream Management*, 1st ed.; Springer: Berlin, Germany, 2016.
53. Drachen, A.; El-Nasr, M.S.; Canossa, A. Game Analytics—The Basics. In *Game Analytics: Maximizing the Value of Player Data*; El-Nasr, M.S., Drachen, A., Canossa, A., Eds.; Springer: London, UK, 2013; pp. 13–40.
54. Nacke, L.E.; Klauser, M.; Prescod, P. Social player analytics in a Facebook health game. In Proceedings of the HCI Korea, Seoul, Korea, 10–12 December 2014; Hanbit Media, Inc.: Seoul, Korea; pp. 180–187.
55. Bernhaupt, R.; Mueller, F.F. Game User Experience Evaluation. In Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems, San Jose, CA, USA, 7–12 May 2016; ACM: New York, NY, USA; pp. 147–148.
56. De Smith, M.J.; Goodchild, M.F.; Longley, P.A. *Geospatial Analysis. A Comprehensive Guide to Principles, Techniques and Software Tools*, 5th ed.; The Winchelsea Press: Winchelsea, UK, 2015.
57. Drachen, A.; Canossa, A. Evaluating motion: Spatial user behaviour in virtual environments. *Int. J. Arts Technol.* **2011**, *4*, 294–314.
58. Hoobler, N.; Humphreys, G.; Agrawala, M. Visualizing competitive behaviors in multi-user virtual environments. In Proceedings of the IEEE Conference on Visualization, Austin, TX, USA, 10–15 October 2004; IEEE Press: Piscataway, NJ, USA; pp. 163–170.
59. Wallner, G.; Kriglstein, S. A Spatiotemporal Visualization Approach for the Analysis of Gameplay Data. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Austin, TX, USA, 5–10 May 2012; ACM: New York, NY, USA; pp. 1115–1124.
60. Wallner, G.; Kriglstein, S. PLATO: A visual analytics system for gameplay data. *Comput. Graphics* **2014**, *38*, 341–356.
61. ESRI. Map Attack—An Urban Geofencing Game. Available online: <http://web.mapattack.org/> (accessed on 13 September 2017).
62. Martí, I.G.; Rodríguez, L.E.; Benedito, M.; Trilles, S.; Beltrán, A.; Díaz, L.; Huerta, J. Mobile application for noise pollution monitoring through gamification techniques. In Proceedings of the 11th International Conference on Entertainment Computing, Bremen, Germany, 26–29 September 2012; Springer: Berlin, Germany; pp. 562–571.
63. Herold, M.; Couclelis, H.; Clarke, K.C. The role of spatial metrics in the analysis and modeling of urban land use change. *Comput. Environ. Urban Syst.* **2005**, *29*, 369–399.
64. Kremer, D.; Schlieder, C.; Feulner, B.; Ohl, U. Spatial Choices in an Educational Geogame. In Proceedings of the 5th International Conference on Games and Virtual Worlds for Serious Applications, Bournemouth, UK, 11–13 September 2013; pp. 1–4.
65. Narain, S.; Vo-Huu, T.D.; Block, K.; Noubir, G. The Perils of User Tracking Using Zero-Permission Mobile Apps. *IEEE Secur. Priv.* **2017**, *15*, 32–41.
66. Jia, Y.; Dong, X.; Liang, Z.; Saxena, P. I Know Where You’ve Been: Geo-Inference Attacks via the Browser Cache. *IEEE Internet Comput.* **2015**, *19*, 44–53.



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).