

Winnie Soon

**ZOMBIFICATION:
THE LIVING DEAD IN SPAM**

**APRJA Volume 4, Issue 1, 2015
ISSN 2245-7755**

CC license: 'Attribution-NonCommercial-ShareAlike'.

[The Zombie] is a soulless human corpse, still dead, but taken from the grave and endowed by sorcery with a mechanical semblance of life — it is a dead body which is made to walk and act and move as if it was were alive.

— William Seabrook, *The Magic Island*, 1929

Spam appears everywhere on the Internet, from downloaded emails to server-based blogs, forums and social media communications. In 2014, statistics show that the proportion of spam almost reaches 70% of entire email traffic.[1] People might not be aware of this datafied phenomenon because most email systems come with spam filtering software that automatically deletes them or categorises them into a special folder, namely 'Junk' or 'Spam'. As such, spam comes into contact with us in a seamless way, though sometimes it still shows up in our normal inbox folder, bypassing 'intelligent' filtering rules. Spam not only consists of commercial advertisements and enticing titbits, it also comes with peculiar email addresses.[2]

These sender addresses become the identity of spam that show up in an email's inbox. In day-to-day form-filling, from paper to electronic registration workflows, supplying an email address is a mandatory field — equally important as a mobile number — to contact another person. In addition, email addresses come with standard naming conventions; a domain usually belongs to or has a connection with a particular organization or institution. For example, I am a researcher in the Department of Aesthetics and Communication at Aarhus University and therefore the university gives me an email with the domain: dac.au.dk. Sometimes a domain does not only describe the nature of an institution — here the letters 'dac' refer to the departmental name — but

also easily indicates a person's geographical location through the last two characters. When a spam email mixes with other emails that appear in the same inbox, we might think that it is a normal email address that carries a similar structure of metadata — a valid email address that one can reply to. A recipient is usually unaware that the sender address can be easily customized, regardless of its authenticity or whether it exists in a network. Therefore, spammers use new sender addresses to transmit messages, and new identities are created in the network.[3] New spammers are created everyday and therefore we are constantly receiving spam email creating ambiguous effects. On the one hand, sender addresses are actively 'living' and distributed in the network, continually monitored by algorithms; on the other, they consume numerous resources of the network and are regarded as "waste" (Parikka and Sampson 4; Gabrys 67) to be traced and trashed. This repetitive production of the 'living dead' resonates with many films, such as *Night of the Living Dead* (1968) and *The Return of the Living Dead* (1985) in popular culture.

This article explores this notion of the living dead in the context of spam culture. How is spam actively and repetitively produced with different identities? I adopt the term 'zombie' to describe spam because, notably, the concept of zombies has been used extensively in popular culture and entertainment, such as films, games and literature (Boluk and Lenz) to describe the phenomenon of mindless slaves (Seabrook). They are usually situated in an environment that has suffered a viral outbreak with contagious effects (Munz et al; Mahoney; Moore). Critiques have compared zombies to dead labour, such as the slavery in Haiti and the labour in the United States (Fischer-Hornung); that is, the exploitation of labour through the concept of alienation, from Marx's theory (Larsen), and labour

practices in global capitalism (Lauro and Embry). Within the context of spam production, as datafied phenomenon, this paper uses the figure of the zombie to describe the computational and network processes of spam automation, which I call ‘zombification’ — alluding to the broader topic of datafication and its consequences. The assumption here is that life once datafied is zombification.

A reflexive approach towards spam

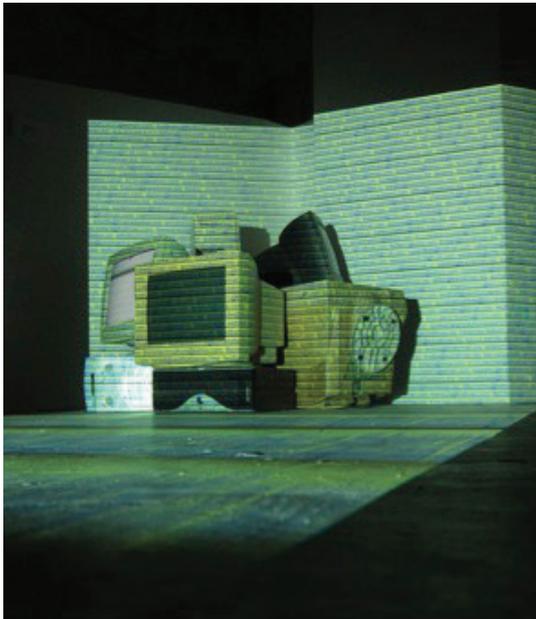


Figure 1: The artwork *Hello zombies* is a network art installation that was exhibited as part of the group exhibition show in Hong Kong with the theme “Tracing Data: what you read is not what we write”. [4] It contains three software programs that constantly refresh and display spammer addresses, sending out spam poems and receiving email replies.

Through my installation *Hello zombies* (2014),[5] I investigate these automations at the level of code, and explore how code interfaces with the mail server to create zombies (see Figure 1). I take a reflexive and artistic

approach to research, paying particular attention to the technical and material aspects. This approach is borrowed from the visual arts where the artistic activity carries the notion of what Sullivan describes as “self-reflexive practice” (110). As such it encompasses multiple methods of inquiry; including reflexive process of observation, interpretation, coding, reading and synthesising of code, text and procedures. Drawing on my personal interest in writing computer code and in software studies, the method departs from the thinking of spam content and literature and operates more in terms of spam production. Following the methodological discussion of revitalising ‘zombie media’, Hertz and Parikka suggest the possibility to discover “new use, contexts and adaptations” to reappropriate unusable technologies (429). How can I reuse spam? My previous collaborative collection of spam poems (a series of literature that is composed entirely from spam content) lends inspiration in this respect. In addition, how can I express the notion of zombification through spam production? Can I compose different spam poems to different recipients, like an automated machine?

With regards to spam, there are a numerous existing discussions: for example, an historical account on how the meaning of spam has changed through technological development (Brunton); a cultural dimension to examine the implications of anomalies such as spam (Parikka and Sampson); artistic methods for spam re-representation (Seiça); the concept of generativity in threatening spam (Zittrain); and the rethinking of spam waste (Parikka and Sampson; Gansing). However, there are few discussions about the cultural aspect of code in spam, in particular the forces through which code interacts with other technical interfaces in spam production. In other words, spam production cannot exist without a programmable machine. The core focus of programmability is based on the

examination of computer code and technical interfaces in order to understand the cultural aspect of an automated machine. Added to this, the notion of zombification constitutes a post-human body (Castillo 167) that follows computer instructions and standard technical interfacing format, producing massive data autonomously and endlessly. However, this data only temporarily exists in the network as it is deleted or blocked by machines or humans. Mutable identity is needed in order to pass through different checking logics and algorithms and to reach the target end. The mutation is achieved through the continuous reading and writing activities of machines. According to Castillo, post-human zombies are without “anima”; they “are animated instead by an outside force” (167) — in this case a programmable machine.

During the coding process, I reflect on the ways in which a spammer captures and composes data from the network. Instead of having a standalone program to compose emails, other interfacing entities such as mail servers and data files are required to compose an email production line. One needs to have a mail server in order to send out high volume emails, but most hosting servers set a strict sending limit per day unless one rents a virtual private server (VPS).[6] An email program is also required to state the parameters that negotiate with a mail server, such as mail domain, username, password, sender address, receiver address, mail subject and mail body. Sending out high volume emails requires reading different data input, and a recipient address, for example, for each email; therefore, a customised program needs to be used, other than a common email client such as Outlook or free internet accounts. However, email marketing is a massive business, and one can pay and rely on companies to provide a sophisticated emailing solution, thereby escaping a complex infrastructure setup.[7] Regardless of

any provided standard software or customised program, composing and sending high volume emails requires computer code that deals with file reading and data processing with a server. As such, code contributes significantly to the process of spam data quantification and automation. However, the role of code cannot be taken for granted from a purely technical perspective in spam production. Instead, these technical structures, the operative dimension and interfacing format allow for a cultural and aesthetic understanding of spam. In other words, the approach is about more than spam as a study object, or questioning what is spam, or the content of spam. Through “the creative inquiry process” (Sullivan 104) of making *Hello zombies*, the actions of my practice and the theoretical reflection are intertwined, mutually informing each other to achieve self-reflexivity.

Mutating parameter value: addresses from senders to receivers

According to Boluk and Lenz, the characteristic of mutation constitutes a zombie as “a force of evolution”, through a “biological model of viral infection” (6). This concept of mutation extends from the biological to a technological model in the twenty-first century, where computer bots and agents are self-modifying through intelligence algorithms and social connectivity to become datafied zombies that invade the network. One of the lists from stop forum spam, an online provider who supplies spammer information, contained around 23,000 spam email addresses for just one day.[8] The list is continuously updated, also with information from network communities. Updated hourly, it is also used in *Hello zombies* to

feed in email addresses to the automated machine. Indeed, spammer addresses can be reported online, and once the address is identified, the email will be added to a block list. Hence, this information will be distributed widely throughout the network. Institutional email systems will then use this list as a base to update their screening processes. According to Spamhaus, an international non-profit organisation that does spam tracking, more than two billion mailboxes are using Spamhaus's blocklists to filter identified spam.[9] In order for an email to enter a mail's inbox folder successfully, not only does the email content need to be customised, the identity has to be carefully considered, so spammers need to change their address in order to escape being caught. This detective mechanism of spam regulates the possibility of its appearance in the client network and the lifespan of a specific spam's identity. But the social life and the distribution of spam is not solely a matter of commercial activities and technical operations, or limited to "a human intentional individual actor" (Latour 7). It is important to also consider the agency of spam (from Latourian actor-network theory), the suggestion being that actor-networks extend "actor — or actant — to non-human, non individual entities" (2). The force of spam comes from the ever-mutating characteristics of spam production and its zombie agency.

Among the spammer email addresses are some real ones, with actual users who are currently using them. Indeed it is fairly common to receive reported cases from the Internet where users' email accounts are being hacked. These hacked mailboxes secretly send spam out as if from actual users.[10] In distributing spam, the field of the email address is easily faked in a computer program: one just has to configure the value of the parameter — sender address, and it can bypass mail servers. Consequently, an automated system will execute this stated

piece of information mindlessly, attaching it to every email that it is going to send out. In other words, there is no checking of the validity of a sender email address, so zombie identities could come with faked and non-identifiable addresses but they also include ones of living persons. However, once they have been tracked, zombies require a new identity to keep the continuity of producing quantified data. Therefore, each zombie identity is paradoxically temporary and generative: the identity keeps mutating over time. This mutating quality is similar to the popular game *Zombie Farm*,[11] where zombies change their body parts to look differently in order to obtain higher success in harvesting. The process is hardly stopped when spammers switch their identities to obtain higher reaching rates as contagions, spreading across the network.

In spam production the configurable parameter, that is the sender address in this case, allows the corresponding value to change easily without impacting the entire production line. It simply replaces a value with another email address. However, this changing parameter value in computer code is not merely a data configuration, but as Neff and Stark put it, also the "information architecture is politics in code" (186). Code, in this emailing context, also includes "technological and social systems" that reshape the value of such email address parameters (Neff and Stark 186). The mutable values have a political condition. The longevity of a zombie's identity is affected by the social demand and the technological development of spam tracking, hacking techniques and security infrastructure. The changing values of such parameters are what Neff and Stark put forward as "political valence" (186).

Similarly, we can also apply the mutating concept into the parameter of a receiver address. This data has enormous commercial value through the reselling of email

addresses. Harvesting live data with active email addresses is arguably one of the most challenging parts for massive emailing. Security is continuously enhanced in email system and filtering rules, and the web checking logic that differentiates robots and humans is becoming more sophisticated. Computer agents, such as web crawlers and web bots, use different ways such as web data mining,[12] spoofing attacks and dictionary attacks to harvest valid and close-to-live addresses. On some occasions a real email address is stolen through spoofing attacks, where spammers “get names and addresses through compromised email accounts, which give them access to contact lists” (Yeaton). Whilst in dictionary attacks, spammers use obsolete and invalid addresses to generate a new recipient address, that is, close-to-live, by slightly amending the username and replacing the old email domain (such as the change of email address from james1@hinet.net to james@hotmail.com) (Clayton).

The value of the receiver parameter stands for an actual target, and it is constantly mutating at code level (see Figure 2). According to common knowledge, one could input more than one target recipient in the carbon copy (cc) or blind carbon copy (bcc) field of an email client interface. Nevertheless, an email server follows a protocol specification that processes addresses one by one through command-line communication in the form of code.[13] The specification “prescribes how the data should be formatted, the type of data allowed” (Hall 13). This is what Alexander Galloway refers to as “network control” (xix). He explains, a “computer protocol is a set of recommendations and rules that outline specific technical standards”. On the one hand, these are technical standards; on the other hand, these “govern the set of possible behavior patterns” as “regulations” (6-7). Regardless of the sender address, a mail server will check the validity of a receiver’s

address. The checking by mail servers includes the validity of the domain, the receiver address, the sending limit and so forth.[14] At the operational level of code, executing such spamming programs means submitting data for an email server’s regulatory check. As such, code cannot escape from the process of network control. In view of the receiver parameter, email servers constantly receive different lists of emails through coding interfaces. These addresses are mutating at the level of code based on the receiver addresses that are found from computer agents. What I want to suggest here is that it is crucial to understand how a program and a system works in order to examine the mutability of code. The parameter of an email address is more than the actual value of it (in the form of numeric and alphabetical value). Indeed, this mutable quality constitutes the entire production chain of spam as, I argue, it is not simply a data configuration that substitutes any data and any value of a parameter. It also contains other cultural significance such as regulatory control and social connectivity, as mentioned. Furthermore, only a receiver address has validity checks while the sender address does not. This loophole facilitates the generation of mutating identities as email addresses in the network. Computationally, it is the mutable quality that allows the parameter value to be changed. When a programmable machine keeps processing scripts and programs, it becomes automated while it is constantly producing quantified data. This undeadness of automation is part of spam culture.

```

try: # sometimes connection fails
    server = smtplib.SMTP('smtp')
    server.login('anonymous@hinet.net', 'xxxxxx')
    server.sendmail(FROM, TO, f'Subject: Hello zombies!')
    send_time = strftime("%m/%d/%Y %H:%M:%S", time.localtime())
    server.quit()
    display = f'Hello to who-email just sent (No zombies - since N3)'
    print (display) #display on screen
    count = 1
    time.sleep(3)
except Exception, errorMsg:
    error_handling(errorMsg)
    
```

Figure 2: A screen shot of the *Hello zombies* program that highlights the variables of sending out an email. The fields ‘FROM’ and ‘TO’ are two parameters where any address value can be configured.

The undead writing of automation

With respect to spam production, it does not come from one machine: many of them are running continuously in the Internet, generating quantified data like a zombie herd. Ratliff describes it as the process of “herding”, where a massive number of zombies receive code instructions to invade the network. Zombies do not have a physical body but they possess a temporal identity and a body of text. They may not survive for long but even if one is being trashed, there are still many others around the network. Boluk and Lenz draw upon Lauro, Embry and Weinstock to discuss the zombie as “a figure of undead labor and consumption” that “is simultaneously a figure of pure automation, of programmed memory that infinitely loops” (7). Zombies are regarded as undead because the automated process minimises human intervention and optimises labour practices. All the digital labourers, such as computer agents and computer job schedulers (also known as ‘cron jobs’) have contributed significantly to the process of automation. Email scheduling can be set in advance to wait for a more effective time to be distributed. In addition, email lists can be continuously fed into the programmable machine. Once the structure is set up, computational parameters, such as sender address, recipient address and body of text, can be mutated in different combinations. As a result, quantified zombies are being distributed from the automated computational process.

Code plays an important role in structuring such an automated system, for example: the flexible and mutable parameters, the ability to interface with other systems and the infinite looping process of high volume data. Drawing upon earlier

concepts of object-oriented programming language from Kristen Nygaard, inventor of the first object-oriented programming language,[15] programming is not only a matter of solving technological problems but is “intended to simulate complex real-world systems” (Lundby 8). Regarding Nygaard’s earlier invention of *Simula* (in the 1960s), many of the concepts, such as schedulers and automatic memory management, are still used in programming languages, which supports concurrent scheduled production and automatic garbage collection that is no longer in use by a program. These enable an efficient and smooth running program as well as a streamlined writing process. Therefore, the structure of programming is fundamentally facilitating the undead processes of automation.

A program contains coding instructions, and these instructions become a score for a machine to run and execute. The automated spam production is also understood as a repeatable writing process, where machines write and perform; according to Wendy Chun, “[No] matter who wrote it or what machine it was destined for; something that inscribes the absence of both the programmer and the machine in its so-called writing” (42). As spam text is generated through computation, we could, therefore, also say code writes spamming emails. From a confining process of computation to a wider framework of capitalism, spam increasingly appears in different sectors and advances its functions. Spammers not only send out bulk messages to promote commercial activities, they also collaborate with hackers “to attack networks, destroy cyber infrastructure, hijack computers, spy on private/confidential data, obtain privileged information (for example: weaponry, industrial secrets, identity theft, other classified information)” (Potdar et al 826). If one of the characteristics of the zombie is the notion of undead, this assertion does not

only describe the nature of its programmable emergence, it is also about how to obtain infinite transactional data through digital consumption such as different email address harvesting methods, that have been mentioned. Digital consumption and production are highly related to networked capitalism, and these two levels of undeadness are, indeed, intertwined.

The reflexive practice of *Hello zombies* examines the notion of the living dead in a speculative way. Setting up an automated production line in *Hello zombies* includes writing computer scripts to fetch spammers' email addresses automatically (these email addresses become the recipients in the artwork); producing and distributing a customised spam poem every eight seconds; reading and extracting any email replies in real time. These computer logics allow a spam email to be fully read by an audience via a screen display, as well as creating a continuous loop to extract new recipient addresses and receive new email. This process-oriented and software-focused approach (without human supervision and direct intervention after code is run) is commonly found in net art and software art practice, such as in *Endless War* by YoHa.[16], and also in *Hello zombies*.

At the practical level of code writing and reading, *Hello zombies* functions and performs according to instructions, but writing code also is a way of self-reflexivity as Chun affirms: "code offers us to think about pleasure, agency, action, danger and, indeed, theory" (Chun, "Codes, Crises and Critical Pleasure"). It reflects upon the datafied condition of both quantified and automated zombies at the level of code. Code associates with data capturing, network protocols and computational processes in the automation of spam production. Thinking about how spam is mutably written enables the understanding of its agency too. Zombies are undead: they are repetitively produced

through different forms of writing: writing to mailboxes and writing for data capturing and processing. Computationally, Chun, however, reminds us that code is a process of "undead writing, a writing that — even when it repeats itself — is never simply a deadly or living repetition of the same" (177). This characteristic of undeadness is not only happening in spam production but also in many other automated network activities that are seamlessly in touch with us, like spam, to produce quantified data in the network. They use a personalised and customised approach to invade the network. In fact, most of them do not have a real identity but we are cohabiting with them. This undeadness — or zombification — suggests attention to the material level of code and the corresponding automated processes in a wider cultural context where things exist temporarily but are constantly reproduced in the network without any real identity.

Zombification in software culture

Zombification describes computational processes of production, addressing the mutable quality of automation. Spam consists of mutating identities. It is continuously and seamlessly produced yet temporarily exists in the network through computation. This temporal existence of the living dead, as I argue, encompasses code automation — an undead and repetitive writing process where a parameters' value is constantly mutating. However, zombification does not only examine the technical dimension of computational processes. This paper tries to articulate the mutable quality at the coding layer, examining its surrounding forces, such as the interface format of a mail server and

an email address, the consumption techniques of email addresses, the parameters and values of a software program, and the repetitiveness and undeadness of writing. Thinking from such material and technical aspects of spam, particularly mutability, we gain a better understanding of spam culture that is associated with its mutating identity, including regulatory controls, loopholes, labour practices, digital consumption and datafication. The computational process of such automated production is part of spam culture that has been somewhat overlooked. Production of spam entails not only automation but also the characteristic of mutability. Through the artwork *Hello zombies*, the critical and aesthetic possibilities of zombification are demonstrated to address the ever-changing datafied phenomenon of digital culture. Indeed, the idea of zombification could be extended to other kinds of software activities that produce quantified data through automated, mutable and programmable machines for qualitative ends.

Acknowledgements

Thank you to *Writing Machine Collective* and my collaborator Susan Scarlata for a chance to exhibit and work on my project *Hello zombies*. I would like to thank my supervisors and the Transmediale workshop participants for reading an earlier draft of this paper and providing helpful criticism and suggestions for improvement. This paper, and my attendance at the Datafied research conference and workshop, were made possible by funding from the Participatory IT research centre at Aarhus University. I am also thankful to the School of Creative Media at the City University of Hong Kong that hosted me as a visiting researcher and enabled me to finish this article there.

Notes

[1] Spam report (2014): <http://securelist.com/analysis/monthly-spam-reports/58559/spam-report-february-2014/>.

[2] Examples of such email addresses are naomiwhitfield274@trash-mail.com and *****@gmail.com. Spam email addresses can be found in stop forum spam: <http://www.stopforumspam.com/downloads/>.

[3] However, many of the email addresses do not exist in the network and are easily identified as spammers. The sender address appears to stand as a proper identity and as such is ready for others to reply to.

[4] The concept statement of the exhibition, *Tracing Data*: http://www.writingmachine-collective.net/wordpress/?page_id=76.

[5] The work *Hello zombies* requires an Internet connection that runs several customised programs on the fly. It examines “nonhuman zombies as a cultural phenomenon that produces quantified data and network identities” and “the living dead that bring forward social, technical, capitalistic and aesthetic relations in everyday lives.” (Writing Machine Collective, 2014) The work responds critically and aesthetically on the wider notion of automated writing and reading in a digital art context. Details on *Hello zombies* can be found at <http://www.siusoon.com/home/?p=1273>.

[6] Details about setting up an email system for mass distribution: <http://www.slideshare.net/anissasimpson900/setting-up-a-email-marketing-system>.

[7] Examples of companies who provide email marketing solutions: Amazon SES: <http://aws.amazon.com/ses/details/> and MailChimp: <http://mailchimp.com/pricing/high-volume-sender/>.

[8] A snapshot as of 17 September 2014. The updated list can be found here: www.stopforumspam.com/downloads/listed_email_1.zip.

[9] The figure is as of Sept 2014. See the Spamhaus project here: <http://www.spamhaus.org/organization/>.

[10] See some examples of reported cases here: <http://askleo.com/why-does-my-account-keep-sending-out-spam/> and <http://lifehacker.com/5875848/how-can-i-find-out-why-my-email-account-just-spammed-my-friends-and-family>.

[11] Zombie farm is a mobile game. See the link here: <http://zombiefarmgame.com/>.

[12] See different ways of harvesting email addresses: <http://www.private.org.il/harvest.html>.

[13] See the full document of SMTP mail server protocol specification here: <http://www.ietf.org/rfc/rfc2821.txt>; A SMTP command ‘RCPT’ is required to communicate with mail server.

[14] See the list of SMTP error messages here: <http://www.serversmtp.com/en/smtp-error>.

[15] Nygaard defines Object-Oriented Programming from the following perspective: The computing process is viewed as the development of a system, consisting of objects (components), through sequences of changing states. See his earlier article: Nygaard, Kristen, 1986: “Program Development as a Social Activity”, *Information Processing* 86, H.-J. Kugler (ed.), Elsevier Science Publishers B.V. (North Holland), IFIP, 1986 (Proceedings from the IFIP 10th World Computer Congress, Dublin, Ireland, September 1-5, 1986), 189-198.

[16] In 2014, *Endless War* was exhibited together with *Hello zombies* in Hong Kong as part of the Writing Machine Collective (5th edition) on the theme *Tracing Data*: <http://www.writingmachine-collective.net/wordpress/?p=489>.

Works cited

- Boluk, Stephanie and Lenz, Wylie. *Generation Zombie: Essays on the Living Dead in Modern Culture*. Mcfarland, 2011. Print.
- Brunton, Finn. *Spam: A Shadow History of the Internet*. The MIT Press. 2013. Print.
- Castillo, David R. "Monsters For the Age of the Post-Human" In *Writing Monsters: Essays on Iberian and Latin American Cultures, Hispanic Issues On Line* 15 (2014). Web. <http://hispanicissues.umn.edu/assets/doc/09_Castillo.pdf>
- Chun, Wendy. "Codes, Crises and Critical Pleasure." Online video clip. Vimeo, 23 Jul. 2010. Web. <<http://vimeo.com/16328263>>
- Chun, Wendy. *Programmed Visions: Software and Memory*. The MIT Press. 2011. Print
- Clayton, Richard. "Stopping Spam by Extrusion Detection" In *First Conference on Email and Anti-Spam (CEAS)*, Mountain View, CA. (2004). Web. <www.cl.cam.ac.uk/~rnc1/extrusion.pdf>
- Cukier, Kenneth Neil and Schoenberger, Viktor Mayer. "The Rise of Big Data: How It's Changing the Way We Think About the World". *Foreign Affairs* (2013). Web. <<http://www.foreignaffairs.com/articles/139104/kenneth-neil-cukier-and-viktor-mayer-schoenberger/the-rise-of-big-data>>
- Fischer-Hornung, Dorothea. "Occupied Bodies: Zombies in William Seabrook's Memoir 'The Magic Island' and the Film 'White Zombie'" Paper presented at the annual meeting of the American Studies Association Annual Meeting, Renaissance Hotel, Washington D.C., <Not Available>. Web. <http://citation.allacademic.com/meta/p315353_index.html>
- Gabrys, Jennifer. *Digital Rubbish: a natural history of electronics*. The University of Michigan Press, 2013. Print.
- Galloway, Alexander R. *Protocol: How Control Exists after Decentralization*. The MIT Press, 2006. Print.
- Gansing, Kristoffer. "'Press Delete' The Politics and Performance of Spamculture." ISEA2010 Conference Proceedings, (2010) 364-265. Web. <http://www.isea2010ruhr.org/files/redaktion/pdf/isea2010_proceedings_p38_gansing.pdf>
- Hall, Eric. *Internet Core Protocols: The Definitive Guide: Help for Network Administrators*. O'Reilly Media, 2000. Print.
- Hertz, G. & Parikka, J. "Zombie Media: Circuit Bending Media Archaeology into an Art Method." *Leonardo* 45,5 (2012) 424-430. Print.
- Latour, Bruno. *Science in Action: How to Follow Scientists and Engineers Through Society*. Harvard University Press, 1987. Print.
- Latour, Bruno. "On Actor-Network Theory. A Few Clarifications plus More than a Few Complications." *Soziale Welt* 47.4 (1996): 369-81. Web. <<http://www.bruno-latour.fr/sites/default/files/P-67%20ACTOR-NETWORK.pdf>>
- Larsen, Lars Bang. "Zombies of Immaterial Labor: The Modern Monster and the Death of Death." *e-flux* 15(2010). Online.

<<http://www.e-flux.com/journal/zombies-of-immaterial-labor-the-modern-monster-and-the-death-of-death/>>

Lauro, Sarah Juliet and Embry, Karen. "A Zombie Manifesto: The Nonhuman Condition in the Era of Advanced Capitalism." *boundary 2* Spring 2008 35(1): 85-108. Web. <<http://boundary2.dukejournals.org/content/35/1/85.citation>>

Lundy, Knut. "The global mover." In Fjuk, A. & Karahasanovic, A. & Kassboll, J. eds., *Comprehensive Object Oriented Learning*, 2006. Print.

Madsen, Ole Lehrman. "Kristen Nygaard – A.M. Turing Award Winner". *Association for Computing Machinery* 2001. Web. <http://amturing.acm.org/award_winners/nygaard_5916220.cfm>

Mahoney, Phillip. "Mass Psychology and the Analysis of the Zombie: From Suggestion to Contagion." in Boluk, S and Lenz, W. eds., *Generation Zombie: Essays on the Living Dead in Modern Culture*, 2011: 113-129. Print.

Munz, Philip, et al. "When zombies attack!: Mathematical modelling of an outbreak of zombie infection." *Infectious Disease Modelling Research Progress*, 4 (2009): 133-150. Print.

Neff, Gina and Stark, David. "Permanently Beta: Responsive Organization in the Internet Era." in Howard, P.N. and Jones, S. eds., *Society Online: The Internet in Context*, Sage Publications. 2004. Print.

Parikka, Jussi and Sampson, Tony D. *The Spam Book: On Virus, Porn, and Other Anomalies from the Dark Side of Digital Culture*. Hampton Press, 2009. Print.

Peaty, Gwyneth. "Infected with Life: Neo-supernaturalism and the Gothic Zombie." in Wasson, S and Alder, E. eds., *Gothic: Science Fiction 1980-2010*, Liverpool University Press. 2011. Print.

Potdar, Vidyasager et al. "The changing nature of Spam 2.0." *CUBE '12 Proceedings of the CUBE International Information Technology Conference* (2012): 826-831. Print.

Ratliff, Evan. "The Zombie Hunters." *The New Yorker* 10 Oct. 2005. Web. <<http://www.newyorker.com/magazine/2005/10/10/the-zombie-hunters>>

Seabrook, W.B. *The Magic Island*. Blue Ribbon Books, 1929. Print.

Seiça, Álvaro. "Anti-Spam: Reinventing Data." *Parsons Journal for Information Mapping* 4:1 (2012) 1-13. Web. <<http://elmcip.net/node/6566>>

Soon, Winnie. *Hello zombies*. 2014. Web. <<http://www.siusoon.com/home/?p=1273>>

Sullivan, Graema. *Art Practice as Research: Enquiry in Visual Arts*. SAGE Publication. 2010. Print.

Yeaton, Monique. "What is email spoofing all about?" *MIT News*. 6 May. 2013. Web. <<http://newsoffice.mit.edu/2013/email-spoofing-whats-it-all-about>>

Zittrain, Jonathan. *The Future of the Internet – And How to Stop It*. New Haven, London: Yale University Press. 2008. Print.